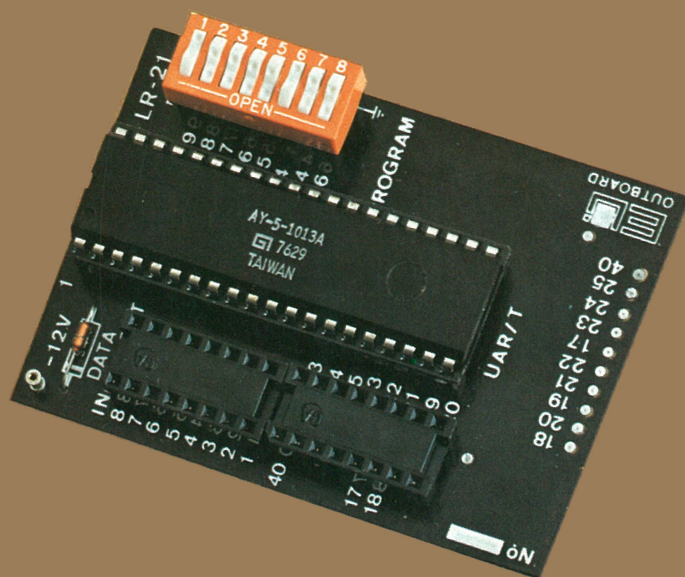
il BUGBOOK® IIA

ESPERIMENTI DI INTERFACCIAMENTO E TRASMISSIONE DATI UTILIZZANTI IL RICEVITORE/TRASMETTITORE UNIVERSALE ASINCRONO (UART) ED IL LOOP DI CORRENTE A 20 mA.

EDIZIONE ITALIANA

David G. Larsen
Peter R. Rony

JACKSON ITALIANA EDITRICE

IL GRUPPO DI BLACKSBURG

I circuiti integrati a larga scala o "chips" LSI stanno creando una seconda rivoluzione industriale che ben presto ci coinvolgerà tutti. La velocità degli sviluppi in questo settore è enorme e diviene sempre più difficile stare al passo coi progressi che si stanno compiendo.

E' sempre stato nostro obiettivo, come Gruppo di Blacksburg, creare tempestivamente e concretamente materiali didattici ed aiuti tali da permettere a studenti, ingegneri, tecnici, ecc. di sfruttare le nuove tecnologie per le loro esigenze particolari. Stiamo facendo questo in molti modi, con libri di testo, brevi corsi, articoli mensili di "computer interfacing" e attraverso la creazione di "hardware" didattico.

I membri del nostro gruppo hanno creato la loro sede a Blacksburg, fra le montagna Appalache del sud-ovest Virginia. Mentre era in corso di preparazione attiva la nostra collaborazione di gruppo, i membri si sono occupati di elettronica digitale, minicomputer e microcomputer.

I nostri sforzi in Italia nel campo didattico sono stati:
- Introduzione, avvenuta nel 1976, sul mercato Italiano da parte della Microlem s.p.a. (Milano) della nostra linea di moduli basati sul sistema di breadboarding senza saldature, o moduli OUTBOARDS®, che facilitano la progettazione e la prova dei circuiti digitali rispetto ai sistemi tradizionali.

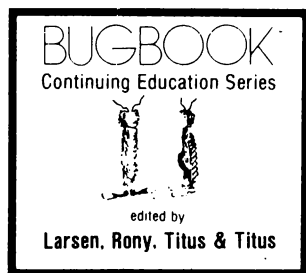
- Traduzione e pubblicazione da parte della JACKSON ITALIANA EDITRICE s.r.l., iniziata nel 1978, dei BUGBOOKS® e della collana di libri Blacksburg Continuing Education Series® che comprende una ventina di titoli riguardanti: l'elettronica di base, microcomputer, convertitori analogico/digitali e digitali/analogici, software per microcomputer, amplificatori operazionali, filtri attivi, phase-locked loops ecc. In ogni libro, oltre al normale testo, vi sono esempi ed esperimenti condotti col sistema passo-passo. Noi crediamo che la sperimentazione consenta di rafforzare i concetti base. Molti titoli stanno per essere tradotti oltre che in Italiano, anche in Spagnolo, Tedesco, Giapponese e Cinese.

- Organizzazione da parte della Microlem, in collaborazione col Virginia Polytechnic Institute and State University e la MIPRO, inizia nel dicembre 1977, di brevi corsi sull'elettronica digitale, la programmazione e l'interfacciamento dei microcomputer. Per l'intera durata dei corsi i partecipanti utilizzano i moduli OUTBOARDS e il microcomputer MMD-1 per verificare i concetti di elettronica digitale, interfacciamento e programmazione presentati nei Bugbooks V e VI. Gli interessati a questi corsi possono rivolgersi alla segreteria dei "Blacksburg Continuing Courses in Italy" tel. (02) 27 10 465.

- Pubblicazione da parte della JACKSON ITALIANA EDITRICE, inziata nel 1978, di articoli, denominati Column, su "Microcomputer Interfacing" nella qualificatissima rivista ELETTRONICA OGGI. Questi columns appaiono anche in quattro riviste americane e in altre tre riviste di elettronica delle quali una Australiana, una Svizzera e una Sud Africana, raggiungendo circa 1.500.000 lettori ogni mese.

- Collaborazione con la SGS-ATES, iniziata nel 1978, per la stesura di materiale didattico relativo alla programmazione e all'interfacciamento del microcomputer SGS-ATES single-board Z-80. Oltre a ciò siamo stati in grado di fornire un prodotto integrato: prodotto progettato per un materiale didattico e contemporaneamente materiale didattico progettato per il prodotto.

- Introduzione da parte dei membri del gruppo di Blacksburg di tecniche didattiche che includono l'uso combinato di stazioni sperimentali multipersone, testi per uso di laboratorio e diapositive 35 mm relative ai testi.
Tutto ciò è stato definito, da alcuni insegnanti italiani, come "il nuovo sistema per la didattica italiana".

Mr. David Larsen e il dr. Peter Rony fanno parte della facoltà dei dipartimenti di Chimica e Ingegneria Chimica del Virginia Polytechnic Institute & State University. Mr. Jonathan Titus e il dr. Christopher Titus fanno parte della Tychon Inc. tutti di Blacksburg, Virginia.

il BUGBOOK® IIA

ESPERIMENTI DI INTERFACCIAMENTO E TRASMISSIONE DATI UTILIZZANTI IL RICEVITORE/TRASMETTITORE UNIVERSALE ASINCRONO (UART) ED IL LOOP DI CORRENTE A 20 mA.

DAVID G. LARSEN
Department of Chemistry

PETER R. RONY
Department of Chemical Engineering

Virginia Polytechnic Institute & State University
Blacksburg, Virginia 24061

Versione italiana

ALDO CAVALCOLI
VALERIO SCIBILIA

JACKSON
ITALIANA
EDITRICE
P.le Massari, 22
20125 Milano

Prima edizione: Febbraio 1979

Stampato in Italia da
Litografia Del Sole - Via Isonzo, 14 - 20094 Buccinasco

SOMMARIO

PREFAZIONE

Nei due precedenti Bugbook, abbiamo imparato come collegare una certa varietà di circuiti integrati della serie 7400. Tali chip, tra cui il 7400, 7402, 7442, 7474, 7493, 74121, 74154, 74181, 74193, ed altri, possono essere considerati l'ABC dell'elettronica. Con questi chip è possibile realizzare praticamente qualsiasi circuito digitale.

Nel presente Bugbook, imparerete come sviluppare circuiti di comunicazione utilizzabili per trasferire informazioni digitali da un vostro circuito ad un qualche sistema di ingresso-uscita, come ad esempio una teletype. Il dispositivo che vi permette di realizzare questa operazione è l'UART, Universal Asynchronous Receiver/Trasmitter, circuito integrato a 40 pin, appartenente alla generazione dei nuovi LSI.

Lo scopo specifico di questo testo è quello di trasferirvi i concetti associati con un tipo di trasmissione dati molto semplice e poco costoso, *la trasmissione dati asincrona seriale.*

Con riferimento alla trasmissione dati, intendiamo la trasmissione di segnali digitali su distanze da pochi metri a molti chilometri. Le tecniche asincrone seriali permettono tali trasmissioni, secondo procedure non complesse. Sono richiesti due UART, uno per ogni terminale della linea di trasmissione, più opportuni circuiti di interfaccia, come quelli tra UART e sistemi a loop di corrente a 20 mA, interfacce per RS-232C, interfacce per modem, linee telefoniche. Ci dedicheremo in modo particolare alle metodologie di interfacciamento tra + 5 volt e 20 mA current loop. Verranno introdotti nella trattazione, alcuni dispositivi utili alla trasmissione dati, come gli Outboard LR-21 UART Outboard,* LR 14 Current Loop Interface Outboard,* LR 13 Line Driver/Receiver Outboard*.

Dalla nostra esperienza, abbiamo verificato la loro notevole utilità nella realizzazione degli esperimenti.

Come nel caso dei precedenti Bugbook, vogliamo ringraziare quanti ci hanno aiutato nella realizzazione, soprattutto i nostri famigliari, per la pazienza e la comprensione dimostrata. Un ringraziamento particolare alla E-L Instruments, che ha realizzato tutti gli Outboard usati negli esperimenti.

David G. Larsen e Peter R. Rony

CAPITOLO 11

INTERFACCIAMENTO E TRASMISSIONE DATI

INTRODUZIONE

Dopo aver acquisito i concetti base dell'elettronica digitale, le caratteristiche di molti circuiti integrati della serie 7400 e le necessarie conoscenze per collegare tali chip l'uno all'altro meglio in modo da realizzare i necessari circuiti, siete in grado di studiare problemi più complessi nel campo dell'elettronica digitale. In questo Capitolo, verrà spiegato come si trasmettono le informazioni da un luogo ad un altro: luoghi che possono essere lontani anche più di un miglio. *La trasmissione dei dati in modo seriale asincrono* in codice ASCII è il metodo più usato per la trasmissione di informazioni digitali: esso non è costoso, è molto versatile, facile da implementare, e può risolvere molti problemi di trasmissione dati che normalmente si incontreranno.

OBIETTIVI

Alla fine di questo Capitolo sarete in grado di:

- Collegare il circuito integrato ricevitore/trasmettitore asincrono universale a 40 pin (UART),
- Dimostrare le funzioni di ciascuno dei 40 pin di un UART,
- Provare il funzionamento in modo half duplex e full duplex dell'UART,
- Costruire un'interfaccia, capace di funzionare sia in modo half duplex che full duplex, tra un circuito integrato UART e una teletype* ASR 33.
- Comparare le caratteristiche del chip standard P MOS UART con i chip C MOS UART IM 6402 e IM 6403.

INTERFACCIAMENTO SERIALE ASINCRONO

Un'interfaccia, può essere definita come il collegamento di elementi di un gruppo (come ad esempio, persone, strumenti), in modo che questi siano in grado di comunicare in modo compatibile e coordinato. Una conoscenza delle basi dell'interfacciamento è diventata importante soltanto recentemente, con l'avvento dell'elettronica digitale e la vendita di molti strumenti digitali con ingressi e uscite diverse. Prima del 1950, quando gli strumenti digitali erano poco diffusi, i problemi d'interfaccia tra due strumenti erano semplici: spesso consistevano nel collegare un paio di fili. Una conoscenza dei concetti relativi all'adattamento delle impedenze, al rumore ed ai loop di terra, era generalmente tutto quello che si richiedeva per realizzare un'interfaccia tra due strumenti appropriati.

Oggi, i problemi dell'interfacciamento tra due strumenti sono stati ingigantiti come conseguenza diretta delle molte nuove possibilità messe a disposizione dei ricercatori e degli ingegneri. Oltre all'interfaccia così detta da analogico ad analogico, che si è già descritta, è ora possibile attuare l'interfacciamento tra strumenti analogici e digitali e, cosa molto più importante, tra strumenti digitali e strumenti digitali o computer. In quest'ultimo caso esistono: a) molti codici digitali per l'interfacciamento (binario, BCD, ASCII, EBCDIC, ecc.), b) molti modi per trasmettere i dati (parallelo, seriale, sincrono e seriale asincrono) e c) la necessità di molti segnali digitali di controllo (indirizzi, codice dispositivo, flag, segnali di abilitazione, comandi di stampa, trigger, clock, segnali di hold, ecc.) che servono a sincronizzare due strumenti che sono "interfacciati", così che essi funzionino "in modo compatibile e coordinato".

Il modo più semplice per interfacciare due strumenti consiste nel collegarli direttamentei (Figura 11-1). Soltanto due o tre fili possono essere necessari, come nel caso prima citato, di trasmissione di dati da analogico ad analogico. Al contrario, nella trasmissione dati digitali sono necessari da 50 a 100 fili a seconda di quanti digit decimali devono essere trasmessi e del numero di segnali digitali di controllo necessari per la sincronizzazione. Non sono stati ancora adottati degli strumenti standard per permettere l'interfaccia di un qualsiasi strumento ad un altro, utilizzando connettori multi-pin standard. Di conseguenza è quasi sempre necessaria una interfaccia per collegare due qualsiasi strumenti di differenti case costruttrici.

Mentre la trasmissione di dati digitali secondo più canali paralleli, è soddisfacente per un semplice laboratorio nel quale tutti gli strumenti sono posti in prossimità l'uno dell'altro, non è più conveniente quando gli strumenti sono posti a molta distanza o l'uno dall'altro, o da un minicomputer maste, da un display, da un data logger o da una stampante. Il costo di un cavo a 60 fili di alcune centinaia o migliaia di metri è veramente proibitivo. In questi casi lo schema mostrato in Figura 11-2, nel quale un convertitore di codice, un ricevitore/trasmettitore, e una linea di trasmissione, sono ora inseriti tra due strumenti, uno dei quali è normalmente un computer, può mantenere l'interfaccia ad un costo modesto.

La linea di trasmissione mostrata in Figura 11-2 è frequentemente una coppia di fili.

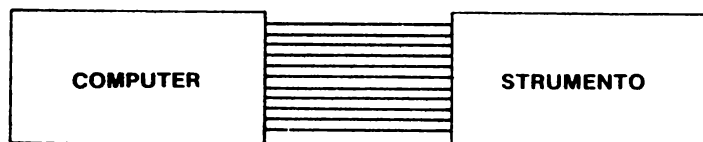



Figura 11-1. Interfaccia tradizionale.

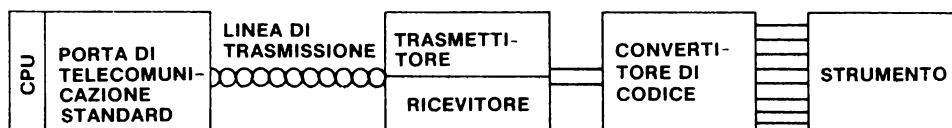



Figura 11-2. Interfaccia per strumentazione che può essere nello stesso laboratorio o ad alcune centinaia di metri di distanza.

intrecciati e schermati per prevenire picchi di rumore. Il metodo di trasmissione dati digitali è chiamato "seriale asincrono" ed è molto spesso utilizzato nel campo delle telecomunicazioni. E' possibile capire meglio la grande diffusione che ha avuto la trasmissione dei dati digitali in modo seriale asincrono, prendendo come punto di partenza la linea di trasmissione, e chiedendoci come è possibile trasmettere al minimo costo dati digitali da un punto ad un altro. Dovremmo probabilmente concludere che un paio di fili ci danno un servizio al minimo costo, in quanto i dati digitali sono spediti sequenzialmente, cioè in modo seriale, piuttosto che in parallelo. D'altra parte il costo di un convertitore di codice (per convertire dati digitali paralleli in seriali) e di un ricevitore/trasmettitore è ben maggiore di quello di un filo di rame, senza contare il lavoro necessario per implementare un cavo a multi-fili.

E' importante notare che il sistema telefonico negli USA, basato su coppie di fili (doppini), fornisce il potenziale per l'acquisizione e il controllo dei dati in tempo reale su distanze di migliaia di chilometri. Sebbene la trasmissione dati seriale asincrona sia molto comune nel campo delle telecomunicazioni, non è molto utilizzata in laboratorio, come ci si aspetterebbe. Noi speriamo che l'automazione dei laboratori attraverso le tecniche seriali asincrone prenda il suo posto al più tardi negli anni 70. Per incoraggiare tali applicazioni, crediamo che tecnici e ingegneri troveranno utile lo sviluppo di questi esperimenti, che mostrano le caratteristiche di una trasmissione di dati seriale asincrona, tecnica che possiede i seguenti vantaggi:

- Basso costo.
- Con gli N MOS UART si raggiungono facilmente velocità di trasmissione dati fino a 1500 dati al secondo.
- Con i C MOS UART si raggiungono facilmente velocità di trasmissione dati di 10.000 dati al secondo.
- Qualsiasi strumento con un'uscita binaria o BCD può essere messo in linea in un sistema a loop chiuso.
- E' richiesta una conoscenza limitata di elettronica digitale. Questo è particolarmente vero con un "ASCII attivo", di recente introduzione, della Nationwide Electronic System (1536 Brandy Parkway, Streamwood®, I11.60103) che è direttamente utilizzabile con una serie di visualizzatori Slimline® e che permette "di collegare fino a 100 pannelli misuratori digitali ad un singolo terminale dati ... utilizzando una semplice coppia di fili!". Egualmente utili sono una serie di moduli ibridi, SERDEX®, commercializzati dalla Analog Devices, Inc. (Route 1, Industrial Park, P.O. Box 280, Norwood, Mass. 02062).
- I linguaggi di programmazione ad alto livello come il BASIC, FOCAL, ed il FORTRAN in tempo reale, sono adatti sia per l'acquisizione che per il controllo dei dati.
- Sono eliminati i problemi legati alla trasmissione analogica dei dati: non vi è nessun problema di rumore e generalmente non sono necessari cavi schermati.
- Non è necessario che l'utilizzatore costruisca l'interfaccia con un minicomputer; canali di connessione diretta all'unità centrale (CPU), possono essere direttamente acquistati ad un prezzo adeguato. Nella loro forma più semplice, due coppie di fili intrecciati connessi all'esistente port della teletype, sono la sola interfaccia necessaria.
- Il sistema è facilmente espandibile, da una semplice acquisizione e controllo di dati on-line ad un sistema che può svolgere molte funzioni su varia strumentazione remota, sottoposta a multiplexer.

- La tecnica è direttamente compatibile con i modem, quindi una qualunque linea telefonica può essere utilizzata per controllo ed acquisizione dati remoti.

Molti strumenti analitici sono attualmente forniti di un'uscita digitale, in particolare alcuni tipi di dispostivi a lettura BCD: nel caso invece di quelli che non possono avere l'uscita analogica digitalizzata, è possibile utilizzare uno dei molti visualizzatori digitali, disponibili, come ad esempio multimetri digitali, visualizzatori per scopi speciali, come lettura della temperatura, della pressione, della velocità di flusso, della forza. Questi strumenti consentono l'utilizzo dei dati nello stesso modo degli strumenti con uscita parallela BCD. Per esempio il visualizzatore "3 1/2 digit" ha un totale di 13 linee di uscita parallele come mostrato in figura 11-3

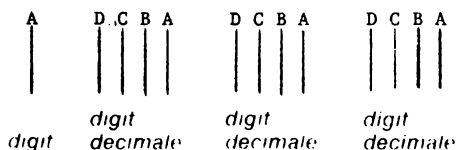



Figura 11-3. Le 13 linee di uscita provenienti da un visualizzatore "3 1/2" digit avente uscite BCD parallele.

Si deve ancora sottolineare che queste sono linee di uscita **parallele:** tutte le informazioni digitali sono utilizzabili simultaneamente. Escludendo la virgola decimale, è possibile "costruire" con un visualizzatore un numero da 0000 a 1999.

E' interessante sapere che il circuito integrato digitale LSI, "trasmettitore/ricevitore asincrono universale, o UART, è molto economico e può raccogliere questi dati BCD e trasmetterli ad un ritmo di mille dati per secondo a un computer, attraverso il collegamento asincrono standard del computer stesso [assumendo da 10 a 12 bit la lunghezza dei dati, equivalente a tre digit decimali interi]. Una tale velocità può soddisfare il 98% di tutte le necessità di elaborazione dati e di controllo in un laboratorio. Il chip è conosciuto come AY-5-1012 nel catalogo della General Instrument Corporation e TR 1402A in quello della Western Digital Corporation. Il chip C MOS UART della Intersil è l'IM 6402/6403. Vi sono anche altri diversi produttori.

La figura 11-4 riassume quanto si è detto . Convertendo dati BCD paralleli in dati ASCII seriali, è possibile interfacciare uno strumento della DEC, Honeywell, NCR, IBM, Control Data, Univac, Burroughs, Data General, ecc. computer o minicomputer e perfino un microprocessor Intel o Texas Instruments.
La figura 11-5 mostra schematicamente come uno strumento di laboratorio può essere interfacciato non solo con un minicomputer, ma da un calcolatore, da microprocessori, terminali in time-sharing, modem ecc. Si capisce ora perchè abbiamo insistito sui concetti della trasmissione e ricezione. (Esperimento N. 7 nel Capitolo 10).

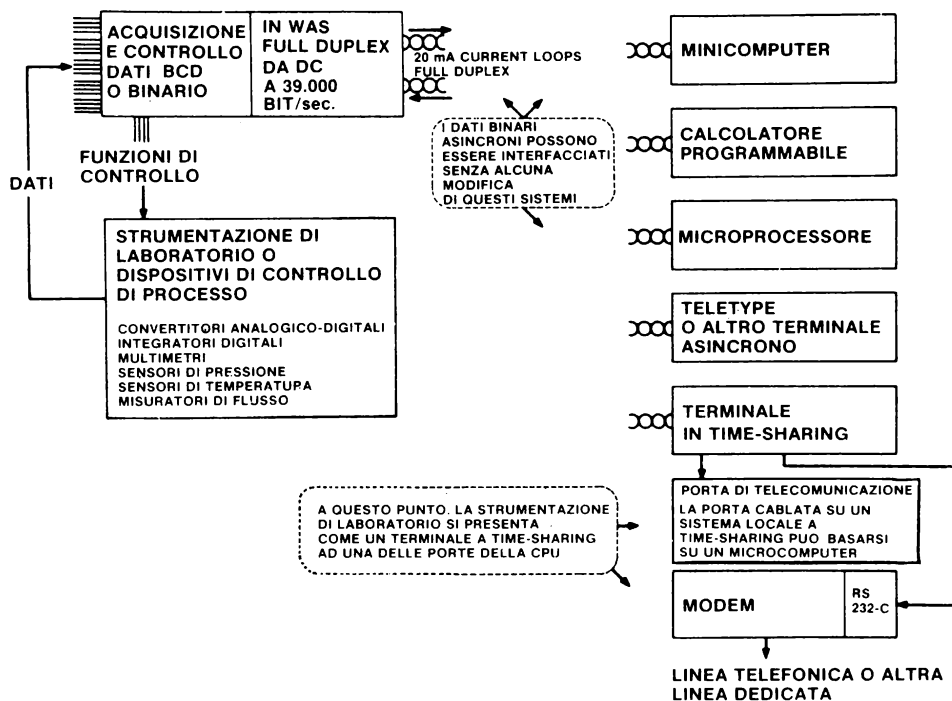



Fig. 11-4. Interfaccia per dati seriali asincroni in codice

IL TRASMETTITORE/RICEVITORE UNIVERSALE ASINCRONO (UART)

L'UART è costituito da un ricevitore a 8 bit asincrono completo e indipendente, e da un trasmettitore completo e indipendente, situati in un circuito integrato a 40 pin. I circuiti integrati LSI sono per lo più costituiti utilizzando una tecnologia silicon gate. L'uscita del trasmettitore è *tri-state,* 1 logico = + 5 Volt, 0 logico = 0 Volt, cioè ad alta impedenza quando è disconnessa. Questo consente di collegare l'uscita ad un altro dispositivo di uscita su un unico sistema di bus, cioè le uscite dell'UART possono essere connesse insieme con un unico filo.

Le velocità dei dati per gli UART standard variano dai valori della corrente continua (DC) fino a 40.000 bit/secondo. L'ingresso di clock è 16 volte la velocità dei dati o al massimo una frequenza di 640 kHz. L'LR-5 Clock Outboard può operare ad una frequenza di clock di 200 KHz, che corrisponde a 200.000/16 = 12.500 bit/secondo. Il trasmettitore e il ricevitore sono indipendenti tra loro e possono funzionare nello stesso tempo e a frequenze di clock differenti. Il bit rate per gli UART C MOS va da DC a 250.000 bit/sec.

La sezione di controllo programma il ricevitore e il trasmettitore ai seguenti stati:

- 5, 6, 7 o 8 bit di dati
- 1 o 2 bit di stop
- parità pari o dispari
- parità o disparità

Il trasmettitore e il ricevitore hanno anche due buffer che danno le costanti di tempo minime relativamente alla abilitazione dati in ingresso e uscita dal sistema. Questo significa che c'è un intervallo di tempo per ogni carattere seriale, nel quale i dati devono essere sottoposti a strobe nel buffer di trasmissione, oppure fuori dal buffer di ricezione, per permettere al sistema di operare alla sua massima velocità, senza perdere i dati. Ciò vi sarà mostrato in uno dei nostri esperimenti, osservando che cosa succede al pin 22 del chip. In trasmissione, quando il pin 22 è allo stato logico1, significa che nel buffer si può caricare un ulteriore "carattere", che potrebbe essere costituito da un numero compreso tra 5 e 8 bit. Se noi forniamo uno strobe al pin 23 (cioè, lo mettiamo per il momento a massa) mentre il pin 22 è allo stato logico 1, osserveremo che è possibile memorizzare un ulteriore carattere. Una volta che è stato memorizzato, il pin 22 passerà allo stato logico 0 e vi rimarrà finchè il buffer potrà ancora accettare un nuovo carattere. Questo è il significato del termine "doubled buffered".

Il trasmettitore e il ricevitore hanno i loro flag, che sono in realtà pin di uscita che cambiano stato logico quando avviene qualcosa di importante nel chip. In questo caso, i "flag" dichiarano lo stato dei registri del buffer di trasmissione e di ricezione, come abbiamo discusso nel paragrafo precedente.

Il ricevitore ha un circuito error-checking, (rilevatore di errori), che può segnalare i seguenti stati:

- Rilevazione di errori hardware
- Rilevazione di overrun del buffer dei dati di uscita
- Rilevazione di errori di parità

Cercheremo di spiegare meglio il significato di tali termini, come "parità", e "flag", negli esperimenti. "Errori hardware" e altri termini simili non sono così importanti e possiamo quindi ignorarli per il momento. Il chip UART è vicino ai livelli di sofisticazione che possono essere ottenuti con le tecnologie LSI, ma sarà facile mostrare che, anche se ha tutte le difficoltà inerenti a un chip a 40 pin, è uno strumento abbastanza semplice da comprendere ed utilizzare.

DESCRIZIONE DEL CHIP UART P MOS

Il chip UART è mostrato in figura 11-5, in cui sono mostrati i pin organizzati sulla base delle loro funzioni. I 40 pin sono stati divisi in tre categorie: alimentazione, trasmissione e ricezione.

Alimentazione:

pin 1: + 5 Volt

pin 2: − 12 Volt (necessario **solo** per la versione P MOS dell'UART)

pin 3: 0 Volt (massa)

L'LR-1 Power Outboard produce soltanto una tensione di + 5 Volt; resta il problema di generare una tensione di − 12 Volt. Si utilizzano batterie a 6 Volt, che sono collegate come mostra la figura 11-6. Due batterie e un Outboard di alimentazione connessi in quel modo danno la tensione richiesta di − 12 Volt al pin 2 dell'UART.

Trasmettitore:

pin 26 (ingresso): Bit 1 (LSB) nel carattere ASCII asincrono. Questo è il 1° bit trasmesso dopo lo start bit che si trova allo stato logico 0.

pin 27 (ingresso): Bit 2

pin 28 (ingresso): Bit 3

pin 29 (ingresso): Bit 4

pin 30 (ingresso): Bit 5

pin 31 (ingresso): Bit 6

pin 32 (ingresso): Bit 7 (MSB) nel carattere ASCII

pin 33 (ingresso): Bit 8 che è normalmente all'1 logico in un carattere ASCII a 11 bit, che contiene uno "start" bit e due "stop" bit. Questo può anche essere il bit di parità e può selezionare sia la parità dispari che quella pari.

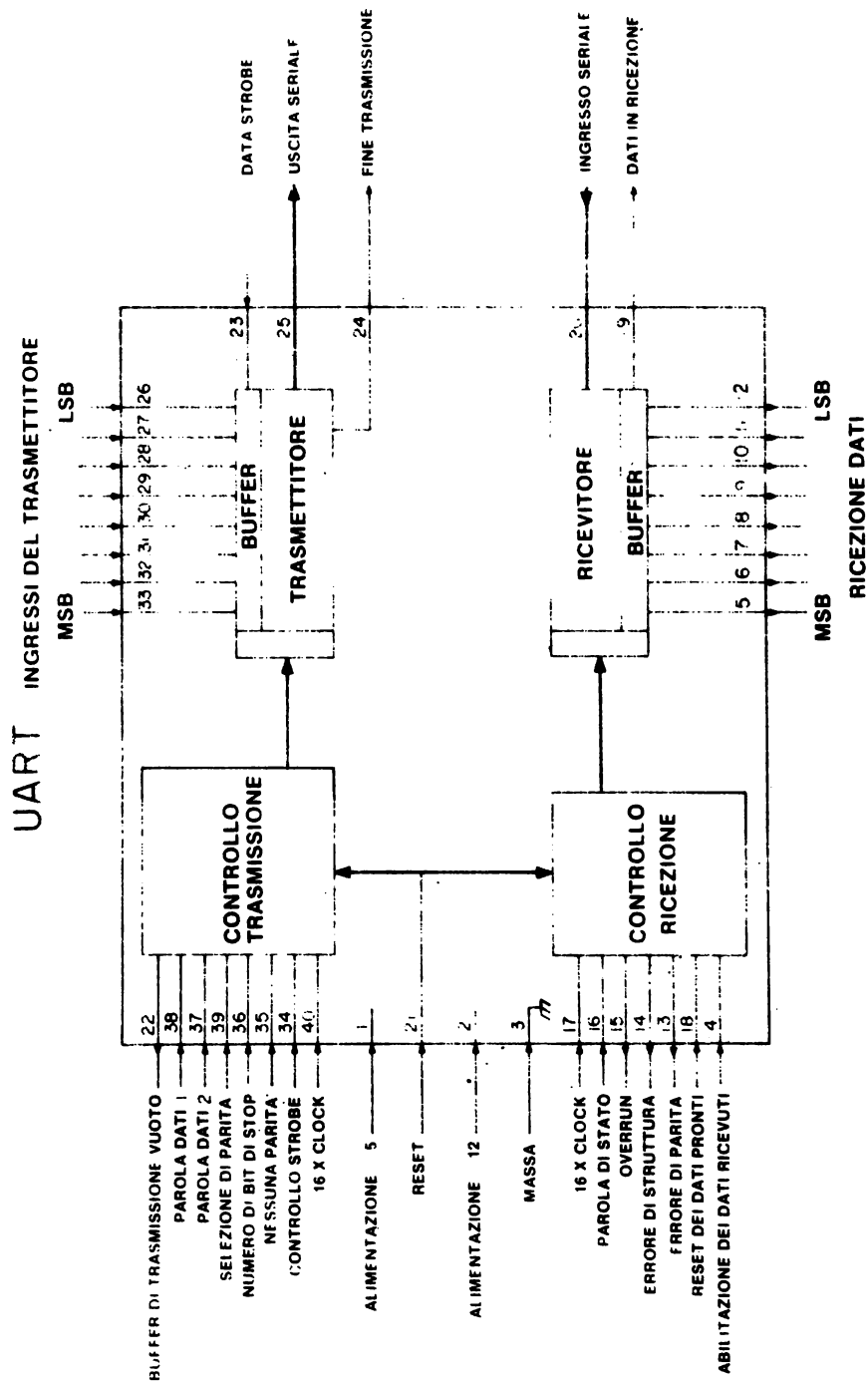



Figura 11-5. Schema a blocchi di un Trasmettitore/Ricevitore Universale Asincrono (UART) a 40 pin.

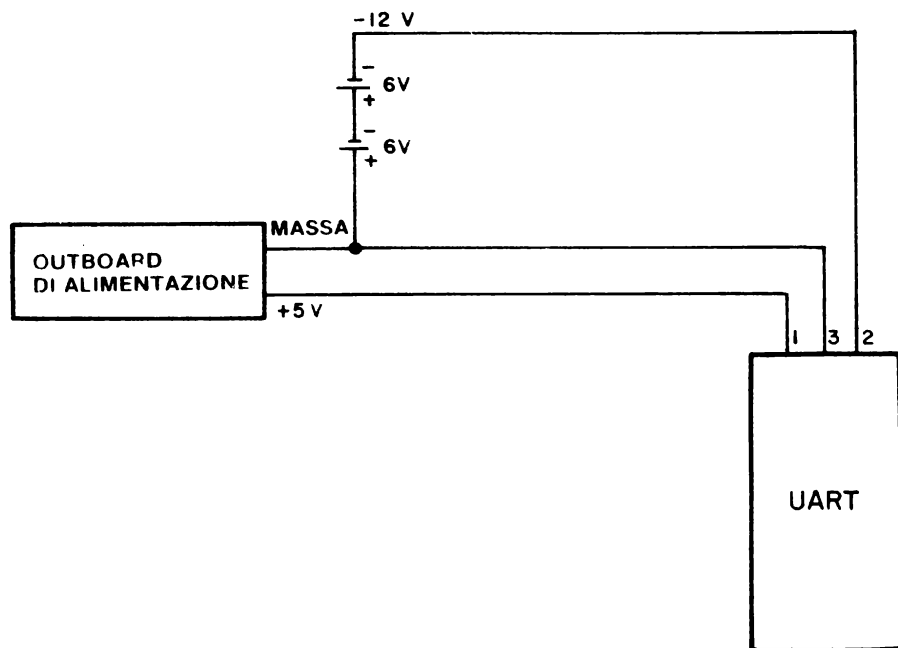



Figura 11-6. L'utilizzo di due pile a 6 Volt per ottenere -12 Volt al pin di ingresso dell'UART. Per la versione C MOS del chip UART non sono richiesti i -12 Volt. Si vedano a questo proposito le indicazioni relative al C MOS UART nella sezione seguente

Tutti i pin precedenti, sono ingressi BCD paralleli, al trasmettitore. Frequentemente gli ingressi BCD sono soltanto dal pin 26 al 29.

pin 40 (ingresso): Ingresso di CLOCK, da 0 a 640 kHz. La frequenza di clock è 16 volte la velocità di uscita dei dati seriali.

pin 25 (uscita): USCITA SERIALE dal trasmettitore: la velocità dei dati varia da 0 a 40.000 bit/secondo (1/16 della frequenza di clock).

pin 22 (uscita): Flag di BUFFER DI TRASMISSIONE VUOTO (trasmitter buffer empty). Se il flag TBMT = 1, il buffer di trasmissione può caricare un carattere. Se il flag TBMT = 0, il buffer è pieno.

pin 23 (ingresso): Un DATA STROBE che passa dall'1 logico allo 0 logico e poi di nuovo all'1 logico inizia la trasmissione di un carattere ASCII. Questo impulso di abilitazione è simbolizzato da ‾|_|‾ , indicante che la linea, inizialmente ad 1, passa per un istante allo stato logico 0.

pin 24 (uscita): Flag di FINE DEL CARATTERE (end of character). Quando il flag di EOC = 0, l'UART è in procinto di trasmettere un carattere al pin 25. Quando il flag EOC = 1 nessun carattere deve essere trasmesso.

pin 21 (ingresso): MASTER RESET. Deve essere allo stato logico 0 per le operazioni del sistema. Allo stato logico 1 azzera tutta la logica.

pin 34 (ingresso): CONTROL STROBE. Un 1 logico attua l'ingresso dei control bit ai pin 35-39 nel registro di holding dei bit di controllo. Questa linea può essere sottoposta a STROBE, _|‾|_ , o può essere posta da hardware allo stato logico 1. Con questo pin abbiamo la possibilità di cambiare in ogni momento i bit di controllo.

pin 35 (ingresso): NESSUNA PARITA'. Uno stato logico 1 eliminerà i bit di parità dal carattere in trasmissione o ricezione. Il bit di stop seguirà immediatamente l'ultimo bit di dati. Se questo pin è utilizzato, deve essere allo stato logico 0.

pin 36 (ingresso): NUMERO DEI BIT DI STOP. Se è allo 0 logico, c'è un solo bit di stop nel carattere. Se è all'1 logico ci saranno due bit di stop.

pin 37 (ingresso):
pin 38 (ingresso): NUMERO DI BIT DEI DATI DEL CARATTERE. La tabella della verità per questi due pin è la seguente:

pin 37	pin 38	Numero di bit dei dati del carattere
0	0	5
0	1	6
1	0	7
1	1	8

pin 39 (ingresso): Il livello logico di questo pin, seleziona il tipo di parità che deve essere immesso immediatamente dopo i bit dei dati. Lo stato logico 0 inserirà una parità dispari. Uno stato logico 1 inserirà una parità pari.

pin 16 (ingreso): ABILITAZIONE DELLA PAROLA DI STATO (Status word enable). Questo pin di ingresso allo stato logico 0 mette i bit della parola di stato sulle linee di uscita ai pin 13, 14, 15, 19 e 22. Questi 5 pin di uscita sono tri-state, ciò a meno che il pin 16 sia allo stato logico 0, che le uscite non siano nè allo 0 logico nê all'1 logico, ma agiscano come se non ci fosse nessuna connessione al momento; cioè ciascun pin ha un'impedenza di uscita molto alta.

Ricevitore:

pin 12 (uscita): Bit 1 (LSB) del carattere ASCII asincrono ricevuto. Questo è il 1° bit ricevuto dopo "start" bit che si trova allo stato logico 0.

pin 11 (uscita): Bit 2

pin 10 (uscita): Bit 3

pin 9 (uscita): Bit 4

pin 8 (uscita): Bit 5

pin 7 (uscita): Bit 6

pin 6 (uscita): Bit 7 (MSB) del carattere ASCII

pin 5 (uscita): Bit 8, che è normalmente all'1 logico in un carattere a 11 bit ASCII, che contiene un bit di start a due bit di stop. Può essere utilizzato anche come bit di parità e può selezionare sia la parità dispari che quella pari.

pin 17 (ingresso): Ingresso di CLOCK, da 0 a 640 kHz per il ricevitore. La frequenza di clock è 16 volte la velocità dei dati dell'ingresso seriale.

pin 20 (ingresso): INGRESSO SERIALE al ricevitore. La velocità dei dati varia ta 0 e 40.000 bit/secondo (1/16 della frequenza di clock).

pin 4 (ingresso): ABILITAZIONE DEI DATI RICEVUTI (received data enable). Uno stato logico 0 sul ricevitore abilita la linea a mettere i dati ricevuti sulle linee di uscita. Le linee di uscita sono costituite dai pin dal 5 al 12.

pin 19 (uscita): Flag di DISPONIBILITA' DEI DATI IN RICEZIONE (receive data available). Si pone all'1 logico quando un intero carattere è stato ricevuto e trasferito al registro di holding del ricevitore. Il pin 4 deve essere allo 0 logico per trasferire i dati ricevuti al registro di holding.

pin 18 (ingresso): RESET DEL FLAG DI DISPONIBILITA' DEI DATI (reset data available). Uno 0 logico resetterà il flag di DISPONIBILITA' DEI DATI IN RICEZIONE al pin 19.

pin 13 (uscita): Flag di ERRORE DI PARITA' IN RICEZIONE (receive parity error). Si dispone all'1 logico se il carattere ricevuto non è in accordo con il tipo di parità, sia pari o dispari, selezionata al pin 39.

pin 14 (uscita): Flag di ERRORE DI STRUTTURA (framing error). Si dispone all'1 logico se il carattere ricevuto non ha alcun bit di stop valido.

pin 15 (uscita): OVER-RUN. Si dispone all'1 logico se il flag di DISPONIBILITA' DEI DATI non è resettato, prima che il carattere presente sia trasferito al registro di holding del ricevitore.

DESCRIZIONE DEL CHIP UART C MOS.

Il ricevitore/trasmettitore universale asincrono C MOS/LSI IM 6402/6403 della Intersil, differisce di poco dalla versione standard P MOS del chip stesso. Nelle pagine seguenti sono forniti alcuni fogli tecnici del costruttore. Per il chip IM 6402, la differenza principale è che il pin 2 non è usato. Vi sono differenze più sostanziali, invece, tra il chip IM 6403 e gli UART standard. Possiamo così sintetizzare:

pin 2	IM 6402:	questo pin non è usato.
	IM 6403:	funziona come ingresso di controllo sia dividi-per-16 che dividi-per-2048 per la frequenza generata dal quarzo collegato ai pin 17 e 40. Uno 0 logico abilita il divisore dividi-per-2048; un 1 logico abilita il divisore dividi-per-16
pin 17 e pin 40:	IM 6403:	usato come ingresso per l'oscillatore a quarzo. Ad esempio, un quarzo di 3,579545 MHz, insieme ad uno 0 logico al pin 2, produce un baud rate di 109,2 Hz, utilizzabile per l'interfacciamento di una teletype.
pin 19:	IM 6403:	non three-state. Sempre attivo.
pin 22:	IM 6403:	non three-state. Sempre attivo.

SOMMARIO DELLE USCITE THREE-STATE DAI CHIP UART.

Le seguenti uscite sono three-state:

pin da 5 a 12:	Le otto uscite dal RECEIVER BUFFER REGISTRE ad 8 bit.
pin 13, 14, 15, 19, 22:	I 5 flag costituenti lo STATUS WORD dell'UART. Quando il pin 16 è all' 1 logico, questi flag sono nel loro stato di alta impedenza. La sola eccezione è per l'IM 6403, pin 19 e 22.

Vogliamo avvertire il lettore di porre attenzione al fatto che gli UART P MOS standard non hanno una risposta sufficientemente veloce per i microcomputer operanti a 2 MHz. Le uscite del buffer dal registro ricevitore devono, infatti, essere sottoposte a buffer con dei buffer addizionali three-state. Questa caratteristica può variare a seconda del fabbricante di UART.

SOMMARIO DEGLI INGRESSI DI ALIMENTAZIONE AI CHIP UART:

pin 1	+ 5 Volt (TR 1602, AY-5-1013, TMS 6012, COM 2017, S 1883) da + 4 a+ 7 Volt (IM 6402 e IM 6403) da + 4 a + 11 Volt (IM 6402A e IM 6403A)
pin 2	—12 Volt (TR 1602 - AY - 5 - 1013, TMS 6012, COM 2017, S1883) Non usato (IM 6402 e IM 6402A) Funzione speciale (IM 6403 e IM 6403A)
pin 3	Massa (tutti i chip UART)

FEATURES

- Operation from DC to 4.0 MHz
- Low Power - typ.<10mW @ 3.0 MHz
- 4V-11V Operation
- Programmable Word Length, Stop Bits and Parity
- Automatic Data Formatting and Status Generation
- Compatible with Industry Standard UART's
- Crystal Operation—IM6403

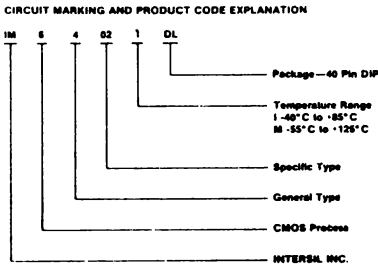
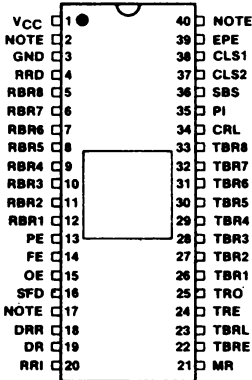
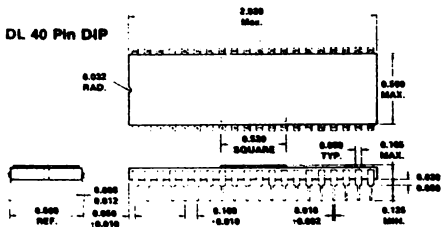
CMOS/LSI UNIVERSAL ASYNCHRONOUS RECEIVER TRANSMITTER (UART) IM6402/6403 IM6402A/6403A

GENERAL DESCRIPTION

The IM6402 and IM6403 are CMOS/LSI subsystems for interfacing computers or microprocessors to an asynchronous serial data channel. The receiver converts serial start, data, parity and stop bits to parallel data verifying proper code transmission, parity, and stop bits. The transmitter converts parallel data into serial form and automatically adds start, parity, and stop bits. The data word length can be 5, 6, 7 or 8 bits. Parity may be odd or even. Parity checking and generation can be inhibited. The stop bits may be one or two or one and one-half when transmitting 5 bit code.

The IM6402 and IM6403 can be used in a wide range of applications including modems, printers, peripherals and remote data aquisition systems. CMOS/LSI technology permits operating clock frequencies up to 4.0 MHz (250K Baud) an improvement of 10 to 1 over previous PMOS UART designs. Power requirements, by comparison, are reduced from 300mw to 10mw. Status logic increases flexibility and simplifies the user interface.

The IM6402 differs from the IM6403 on pins 2, 17, 19, 22, and 40 as shown in the connection diagram. The IM6403 utilizes pin 2 as a control and pins 17 and 40 for an inexpensive crystal oscillator as shown on page 5. TBREmpty and DReady are always active. All other input and output functions of the IM6402 and IM6403 are as described.

ORDERING INFORMATION

CONNECTION DIAGRAM

NOTE:

PIN	IM6402	IM6403
2	N/C	CONTROL
17	RRC	OSC IN
40	TRC	OSC OUT

PACKAGE DIMENSIONS

IM6402/03

ABSOLUTE MAXIMUM RATINGS

Supply Voltage		+7.0V
Input or Output Voltage Applied		GND -0.3V to V_{CC}+0.3V
Storage Temperature Range		-65°C to +150°C
Operating Temperature Range	Industrial IM6402/03I	-40°C to +85°C
	Military IM6402/03M	-55°C to +125°C

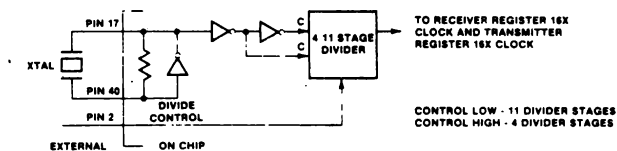
DC CHARACTERISTICS V_{CC} = 5.0 +− 10%. T_A = Operating Temperature Range

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Logical "1" Input Voltage	V_{IH}		V_{CC}− 2.0			V
Logical "0" Input Voltage	V_{IL}				0.8	V
Input Leakage	I_{IL}	$OV \le V_{IN} \le V_{CC}$	− 1.0		1.0	μA
Logical "1" Output Voltage	V_{OH}	I_{OH} = −0.2 mA	2.4			V
Logical "0" Output Voltage	V_{OL}	I_{OL} = 2.0 mA			0.45	V
Output Leakage	I_O	$OV \le V_O \le V_{CC}$	− 1.0		1.0	μA
Supply Current	I_{CC}	V_{IN} = GND or V_{CC}; Output Open		1.0	100	μA
Input Capacitance	C_{IN}			7.0	8.0	
Output Capacitance	C_O			8.0	10.0	pF

AC CHARACTERISTICS V_{CC} = 5.0V, T_A = 25°C

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Clock Frequency IM6402	f_{clock}		D.C.	3.0	2.0	MHz
Crystal Frequency IM6403	$f_{crystal}$		D.C.	4.0	3.58	MHz
Pulse Widths CRL, DRR, TBRL	t_{pw}		150	50		ns
Pulse Width MR	t_{pw}	See switching time	400	200		ns
Input Data Setup Time	t_{SET}	waveforms 1, 2, 3	50	20		ns
Input Data Hold Time	t_{HOLD}		60	40		ns
Output Propagation Delays	t_{pd}			80	120	ns

IM6403 UNIVERSAL ASYNCHRONOUS.RECEIVER TRANSMITTER WITH ON CHIP 4/11 STAGE DIVIDER

The IM6403 differs from the IM6402 on three inputs, TRC, RRC, and pin 2, and two outputs TBRE and DR. Outputs DR and TBRE are not three-state, but are always active.

The divider chain output acts as a 16X clock to both the receiver register and transmitter register. Consequently both receiver and transmitter operate at the same frequency. The TRClock and RRClock inputs are used for a crystal oscillator while pin 2 controls the number of divider stages.

The on chip divider and oscillator allow an inexpensive crystal to be used as a timing source rather than additional circuitry such as baud rate generators. For example, a color TV crystal at 3.579545MHz results in a baud rate of 109.2 Hz for an easy teletype interface.

PIN ASSIGNMENT AND FUNCTIONS

PIN	SYMBOL	DESCRIPTION
1	VCC	+5 Volts Supply
2	IM6402-N/C IM6403-Control	No Connection 4/11 Stage Divider High 4 Stage Low 11 Stage
3	GND	Ground
4	RRD	A High level on RECEIVER REGISTER DISABLE forces the receiver holding register outputs RBR1-RBR8 to a high impedance state
5	RBR8	The contents of the RECEIVER BUFFER REGISTER appear on these three-state outputs Word formats less than 8 characters are right justified to RBR1
6	RBR7	See Pin 5 - RBR8
7	RBR6	See Pin 5 - RBR8
8	RBR5	See Pin 5 - RBR8
9	RBR4	See Pin 5 - RBR8
10	RBR3	See Pin 5 - RBR8
11	RBR2	See Pin 5 - RBR8
12	RBR1	See Pin 5 - RBR8

PIN	SYMBOL	DESCRIPTION
13	PE	A high level on PARITY ERROR indicates received parity does not match parity programmed by control bits. When parity is inhibited this output is low.
14	FE	A high level on FRAMING ERROR indicates the first stop bit was invalid.
15	OE	A high level on OVERRUN ERROR indicates the data received flag was not cleared before the last character was transferred to the receiver buffer register
16	SFD	A high level on STATUS FLAGS DISABLE forces the outputs PE, FE, OE, DR, TBRE to a high impedance state.
17	IM6402-RRC IM6403-OSCIN	The RECEIVER REGISTER CLOCK is 16X the receiver data rate
18	DRR	A low level on DATA RECEIVED RESET clears the data received outputDR, to a low level
19	DR	A high level on DATA RECEIVED indicates a character has been received and transferred to the receiver buffer register
20	RRI	Serial data on RECEIVER REGISTER INPUT is clocked into the receiver register

PIN	SYMBOL	DESCRIPTION
21	MR	A high level on MASTER RESET clears PE, FE, OE, and DR to a low level and sets the transmitter output to a high level after 18 clock cycles
22	TBRE	A high level on TRANSMITTER BUFFER REGISTER EMPTY indicates the transmitter buffer register has transferred its data to the transmitter register and is ready for new data
23	TBRL	A low level on TRANSMITTER BUFFER REGISTER LOAD transfers data from inputs TBR1-TBR8 into the transmitter buffer register A low to high transition on TBRL indicates data transfer to the transmitter register If the transmitter register is busy, transfer is automatically delayed so that the two characters are transmitted end to end
24	TRE 6	A high level on TRANSMITTER REGISTER EMPTY indicates completed transmission of a character including stop bits
25	TRO	Character data, start data and stop bits appear serially at the TRANSMITTER REGISTER OUTPUT
26	TBR1-TBR8	Character data is loaded into the TRANSMITTER BUFFER REGISTER via inputs TBR1-TBR8 For character formats less than 8 bits the TBR8, 7, and 6 inputs are ignored corresponding to the programmed word length

PIN	SYMBOL	DESCRIPTION
27	TBR2	See Pin 26 - TBR1
28	TBR3	See Pin 26 - TBR1
29	TBR4	See Pin 26 - TBR1
30	TBR5	See Pin 26 - TBR1
31	TBR6	See Pin 26 - TBR1
32	TBR7	See Pin 26 - TBR1
33	TBR8	See Pin 26 - TBR1
34	CRL	A high level on CONTROL REGISTER LOAD loads the control register.
35	PI	A high level on PARITY INHIBIT inhibits parity generation, parity checking and forces PE output low
36	SBS	A high level on STOP BIT SELECT selects 1.5 stop bits for 5 character format and 2 stop bits for other lengths.
37	CLS2	These inputs program the CHARACTER LENGTH SELECTED (CLS1 low CLS2 low 5 bits) (CLS1 high CLS2 low 6 bits) (CLS1 low CLS2 high 7 bits) (CLS1 high CLS2 high 8 bits)
38	CLS1	See Pin 37 - CLS2
39	EPE	When PI is low a high level on EVEN PARITY ENABLE generates and checks even parity A low level selects odd parity
40	IM6402-TRC IM6403-OSCOUT	The TRANSMITTER REGISTER CLOCK is 16X the transmit data rate

LA TELETYPE® MODELLO ASR 33

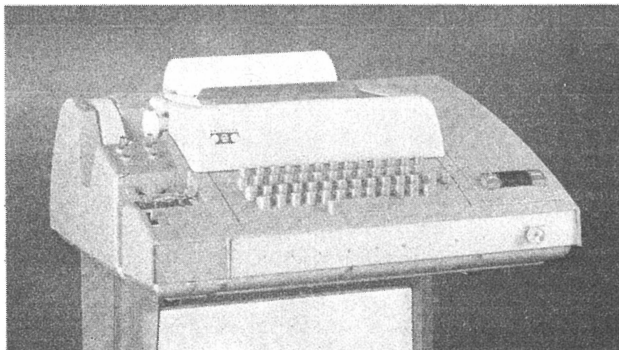
La telescrivente modello ASR 33 costruita dalla Teletype Corporation, è praticamente una macchina da scrivere con una linea di comunicazione che utilizza caratteri seriali asincroni a 11 bit (20 mA currente loop). E' costituita da una tastiera, una stampante, un lettore di banda perforata, e un perforatore di banda. Invece di descrivere la telescrivente, vi sono allegate le pagine di descrizione gentilmente fornite dalla Digital Equipment Corporation (DEC), tratte dal "PDP 8/E & PDP 8/M Small Computer Handbook" (Digital, Equipment, Corporation, Maynard, Massachussetts 1971.) Il più importante controllo sulla telescrivente è lo switch LINE/OFF/LOCAL nell'angolo in fondo a destra. Nella posizione "LINE", la telescrivente entra in funzione e si connette all'input/output di un computer, oppure a dispositivi di I/O. Nella posizione OFF, la telescrivente è inattiva. Nella posizione LOCAL, la telescrivente funziona solo come macchina da scrivere e non può comunicare con un computer. *Normalmente si utilizzerà la telescrivente in posizione LINE.*

Console della Teletype modello ASR33.

Tabella 11-1

Controlli e Indicatori della Telescrivente ASR 33

CONTROLLI O INDICATORI	*FUNZIONI*
Tasto REL	Permette di rimuovere o inserire il nastro perforato.
Tasto B SP	Sposta il nastro di una posizione indietro, permettendo sia correzioni manuali che cancellature del carattere già perforato.
Tasto OFF/ON	Controlla l'uso del perforatore insieme alle operazioni della tastiera/ stampante della telescrivente.
Interruttore START/STOP/FREE	Controlla l'uso del lettore di bande perforate unitamente alle operazioni della teletype. In FREE il lettore è disabilitato, permettendo lo spostamento manuale del nastro nel lettore. In STOP il lettore è inserito ma non alimentato. In START il lettore è abilitato e opera sotto il controllo del programma.

Tabella 11-1 (continua)

Tastiera	Permette di stampare dei caratteri quando è usata come macchina da scrivere, di perforare la banda quando è premuto il tasto ON, o di inviare i dati di ingresso al computer quando l'interruttore LINE/OFF/LOCAL è in posizione LINE.
Interruttore LINE/OFF/LOCAL	Permette di alimentare la telescrivente e di connetterla al computer. In LINE la telescrivente è alimentata e connessa con la linea del computer. In OFF non è alimentata. In LOCAL è alimentata per operazioni non in linea. Sia in LINE che in LOCAL la teletype deve essere alimentata tramite lo switch POWER.

Tastiera della teletype

OPERATIVITA' DELLA TASTIERA
La tastiera è simile a quella di una macchina da scrivere eccetto che per alcuni caratteri che non vengono stampati, messi sotto forma di caratteri maiuscoli. Per stampare caratteri o simboli, $, % * che compaiono nella posizione superiore dei tasti numerici e di alcuni tasti alfabetici, occorre premere il tasto desiderato unitamente al tasto SHIFT. L'indicazione di certe funzioni non stampabili sono mostrate nella parte superiore di alcuni tasti alfabetici. Premendo tali tasti insieme al tasto CTRL (control), tali funzioni sono attivate: la tabella 11-2 mostra alcuni tasti comunemente usati, aventi funzioni speciali nel linguaggio del computer PDP-8/E:

Tabella 11-2

Funzioni Speciali da Tastiera

TASTO	*FUNZIONE*	*USO*
SPACE	Spazio	Serve a delimitare e combinare simboli o numeri in un programma simbolico.
RETURN	Ritorno del carrello	Utilizzato per terminare una linea di un programma simbolico.
HERE IS	Blank sul nastro	Utilizzato come leader-trailer (effettivo solo in LOCAL).
RUBOUT	Cancellatura caratteri	Utilizzato per cancellare dei caratteri;di di fatto perfora tutti i canali del nastro.
CTRL/REPT/P	Codice 200	Utilizzato come leader/trailer di nastri binari (i tasti devono essere rilasciati in ordine inverso: P. REPT, CTRL).
LINE FEED	Ritorno a capo	Permette al rullo di carta di avanzare di una linea e di ritornare sulla 1° posizione di stampa.

OPERAZIONI DI STAMPA
La stampante fornisce una copia stampata dei dati di ingresso e uscita alla velocità di 10 caratteri al secondo. Quando la teletype è in linea (LINE) la copia è generata dal computer; quando é in LOCAL la copia è generata alla premuta di un tasto.

OPERAZIONI DI LETTURA DELLE BANDE PERFORATE
Il lettore di bande perforate è utilizzato come ingresso dei dati perforati sugli 8 canali del nastro alla velocità massima di 10 caratteri al secondo. I tasti di controllo e le loro funzioni sono descritte di seguito:

START	Attivazione del lettore: lo sprocket del lettore è in presa ed operativo.
STOP	Disattivazione del lettore: lo sprocket del lettore è in presa ma non operativo.
FREE	Disattivazione del lettore lo sprocket del lettore non è in presa.

Di seguito sono indicati alcuni numeri, lettere e operazioni della tastiera della telescrivente
ed i rispettivi codici ottali che servono per comunicare con la telescrivente utilizzando un chip UART. La tabella riporta le parole di 7 e 8 bit, in codice ottale, che appariranno sul display a LED a 7 segmenti negli esperimenti di questo Capitolo.

Tabella 11-3. Il codice half-ASCII.

Lettere, numeri simboli o operazioni	*Parola ottale a 8 bit*	*Parola ottale a 7 bit*
0	260	060
1	261	061
2	262	062
3	263	063
4	264	064
5	265	065
6	266	066
7	267	067
8	270	070
9	271	071
A	301	101
B	302	102
C	303	103
D	304	104
E	305	105
F	306	106
G	307	107
H	310	110
I	311	111
J	312	112
K	313	113
L	314	114
M	315	115
N	316	116
O	317	117
P	320	120
Q	321	121
R	322	122
S	323	123
T	324	124
U	325	125
V	326	126
W	327	127
X	330	130
Y	331	131
Z	332	132
.	256	056
,	254	054
?	277	077
=	275	075
*	252	052
$	244	044
%	245	045
!	241	041
'	247	047

space	240	040
LINE FEED	212	012
RETURN	215	015

Il tasto RETURN è un tasto di ritorno del carrello, che termina una linea sulla stampante. Il tasto LINE FEED fa avanzare il carrello di una linea.

Altri trasmettitori/ricevitori, come il tubo a raggi catodici (CRT), hanno le stesse caratteristiche operative della telescrivente purchè operino, nella comunicazione, secondo le modalità TTL, loop di corrente di 20 mA, oppure RS-232C. Un Outboard* di interfaccia può essere necessario tra l'UART e tali dispositivi. A riguardo si veda più avanti in questo capitolo la descrizione degli Outboards*. La velocità di trasmissione dati dell'UART deve essere uguale a quella dei dispositivi con i quali si deve comunicare. Un trasmettitore/ricevitore CRT fornisce una velocità di 110, 220, 330, 440, 880, 1200 e 2400 bit/secondo.

LOOP DI CORRENTE

Nei Bugbooks I e II si è visto che una tensione di + 5 volt corrisponde allo stato logico 1 e una tensione vicina al potenziale di massa corrisponde allo stato logico 0. Il modello di telescrivente ASR 33 comunica con un computer tramite current loop.

stato logico 1: presenza di una corrente di circa 20 mA nel loop di corrente.

stato logico 0: assenza di corrente nel loop di corrente.

I current loop sono delle linee di trasmissione a bassa impedenza che presentano una elevata immunità al rumore. Segnali digitali possono essere trasmessi su circuiti, su distanze più lunghe di un chilometro, senza perdere nessuna informazione.

Uno schema elettrico di un tipico current loop è mostrato nella seguente figura; si può notare che:

- La corrente di 20 mA nel loop è fornita con l'aiuto di un alimentatore di + 5 volt e un regolatore di corrente di 20 mA.
- Le connessioni dei segnali DC sul back panel della telescrivente, sono collegate ai pin 3 e 4 (tastiera o lettore di bande rispettivamente) e ai pin 6 e 7. La tastiera è talvolta conosciuta come "trasmettitore": la stampante come "ricevitore".
- La polarità delle connessioni ai terminali della tastiera o della stampante della telescrivente sono come mostrato in figura.
- Uno o più chip UART possono essere collegati nel current loop. Tuttavia, l'*uscita TTL proveniente da un UART, che è sia a + 5 volt che massa deve essere convertita per operazioni in loop di corrente a 20 mA. Non è infatti possibile, collegare una telescrivente direttamente con un'uscita a + 5 volt, proveniente da un circuito integrato.* Il circuito deve avere un alimentatore indipendente.

Si può concludere che dobbiamo convertire il modo di operare da + 5 volt a quello di 20 mA. Ciò è possibile tramite un Outboard che sarà descritto immediatamente dopo questo argomento.

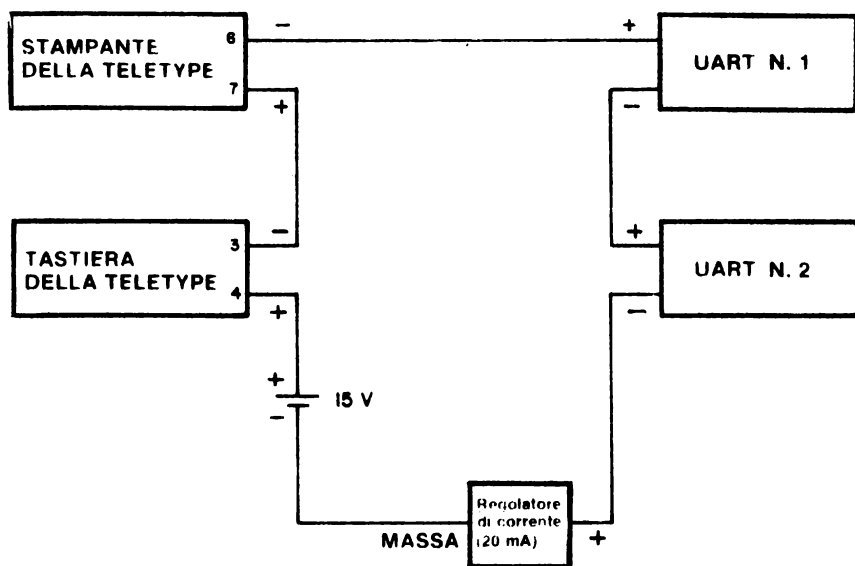



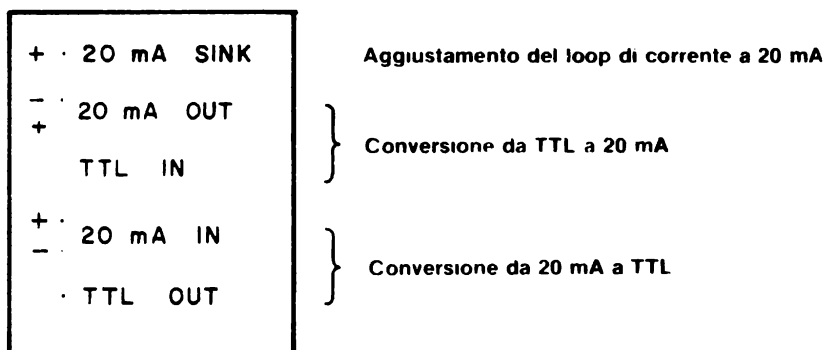
Figura 11-7. Un tipico loop di corrente.

L'OUTBOARD® LR-14 DI INTERFACCIA 14 TTL 20 mA CURRENT LOOP

L'Outboard® LR-14 di interfaccia TTL/20 mA esegue le seguenti funzioni:

- La conversione di un segnale logico di + 5 volt in un segnale logico di 20 mA.
- La conversione di un segnale logico di 20 mA in un segnale logico di + 5 volt.
- La regolazione della corrente di 20 mA che percorre il circuito, data da un'alimentazione a + 15 volt.

Uno schema dell'Outboard® LR-14 è mostrato di seguito. Una importante caratteristica dell'Outboard è l'utilizzo di un paio di *opto-isolatori. Un opto-isolatore è utilizzato nei circuiti elettronici per avere una separazione elettrica tra due parti di un circuito. Un opto-isolatore può essere considerato un relè ottico, nel quale il segnale elettrico è convertito in un segnale ottico con l'aiuto di un LED; il segnale ottico è poi rivelato da un dispositivo a semiconduttore sensibile alla luce, come ad esempio un fotoresistore od un fototransistor. Si può ottenere una tensione di isolamento di 1000 fino a 30.000 volt a secondo del potenziale di breakdown del materiale isolante utilizzato. Deve essere chiaro che l'uso di un opto-isolatore, per isolare circuiti a*+ 5 volt da loop di corrente di 20 mA, è relativamente facile.

Nell'Outboard® LR-14 sono presenti un totale di 7 pin di ingresso/uscita, come mostra lo schema dell'Outboard dato qui di seguito:

Riprenderemo questo argomento nel paragrafo che segue, dove le caratteristiche di questo Outboard saranno chiarite ulteriormente.

ISTRUZIONI PER L'UTILIZZO DELL'OUTBOARD* LR-14

L'Outboard* LR-14 di interfaccia 14 TTL/20 mA loop di corrente, può essere utilizzato per interfacciare un segnale TTL a + 5 volt, come un segnale di ingresso/uscita proveniente da un trasmettitore/ricevitore universale asincrono (UART), con telescriventi, terminali CRT, e altri dispositivi che hanno un current loop di 20 mA, sia half che full duplex.

SPECIFICHE

Le specifiche dell'Outboard* LR-14 sono le seguenti:

- Ingresso/uscita TTL a + 5 volt: fan-in di 1, fan-out di 10;
- Ingresso/uscita di 20 mA: "0" logico - - meno di 3 mA di corrente
  "1" logico - - più di 15 mA di corrente

  La massima tensione di loop all'"1" logico è di 30 volt.
- Gli opto-isolatori consentono un isolamento completo del current loop a 20 mA dal trasmettitore/ricevitore TTL. Ciò è molto importante per eliminare il tradizionale problema dei loop di massa quando diverse parti di un sistema sono poste insieme. Gli opto- isolatori sono protetti da diodi, per la polarità inversa nel loop.
- Un regolatore di corrente sulla scheda consente di regolare una tensione di + 5 volt per fornire una corrente di 20 mA per lo stato logico 1 del circuito. Il regolatore sostituisce il resistore di limitazione della corrente. Il regolatore non è isolato da massa e non può essere usato per un ingresso/uscita isolato.
- La velocità di trasmissione dei dati è la seguente:
  20 k bit/secondo (minimo)
  30 k bit/secondo (tipico)
  40 k bit/secondo (massimo)

Il condensatore di filtro (0,01 μF) sulla piastra, consente al sistema di funzionare a tutte le velocità. la reiezione al rumore può essere incrementata restringendo la larghezza della banda in cui opera il sistema. Ciò si ottiene aggiungendo una capacità esterna ai due jack BP 25 di interconnessione. La velocità della trasmissione dei dati, per i diversi valori della capacità, è la seguente:

Capacità	*Velocità di trasmissione dei dati*
1 μF	da 0 a 300 bit/secondo
0,5 μF	da 0 a 1 k bit/secondo
0,1 μF	da 0 a 10 k bit/secondo
0.05 μF	da 0 a 15 k bit/secondo
nessuna	da 0 al massimo dei k bit/secondo

Uno schema del circuito per le sezioni da 20 mA a TTL, da TTL a 20 mA e per la sezione del regolatore di corrente dell'Outboard*, sono mostrate di seguito.

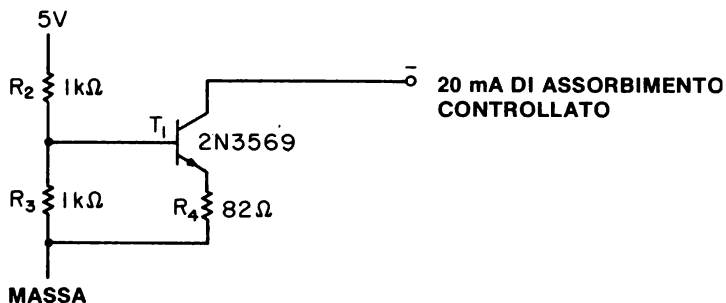
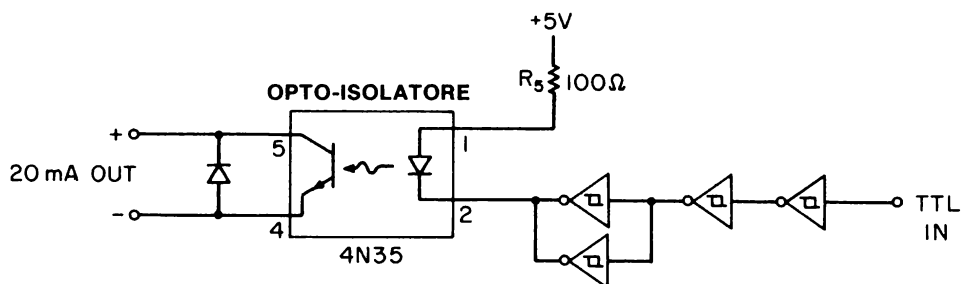
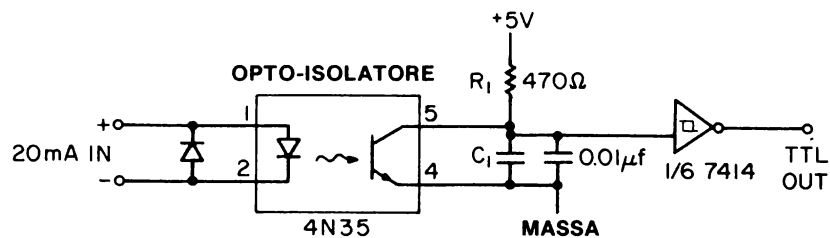



Figura 11-8. Circuiti impiegati dall'Outboard® di interfaccia LR-14.

ISTRUZIONI PER L'UTILIZZO DELL'OUTBOARD® LR-13

L'Outboard® LR-13 Line Driver/Receiver ed il TTL/RS-232C Interface Outboard® possono essere usati entrambi come interfaccia tra un UART e un modem, cioè come line driver/receiver "general purpose" con una velocità superiore ad 1 megabit/secondo. L'Outboard® è costituito da un driver dual line Signetics 8T15 e da un ricevitore dual line Signetics 8T16. Le caratteristiche sono:

- Interfaccia tra un trasmettitore/ricevitore universale asincrono (UART) e un modem (RS-232C) sia in trasmissione che in ricezione.
- Interfaccia tra un qualsiasi altro segnale TTL (+ 5 volt) e un modem, o tra un modem e un ingresso TTL.
- Soddisfa o supera le caratteristiche di un EIA standard RS-232B e C, MIL STD 188B, e CCITT V 24 per line receiver e line driver di comunicazione.
- Agisce come un trigger di Schmitt, un convertitore d'onda sine-to-square, o permette operazioni accoppiate AC.
- Fornisce alta immunità al rumore sia differenziale che in modo comune.
- Può essere utilizzato secondo modalità operative EIA Fail-safe.

SPECIFICHE

Le specifiche dell'Outboard® LR-13 e dell'Outboard® di interfaccia TTL-RS 232C sono essenzialmente simili a quelle applicabili ai Signetics 8T15 e 8T16. Le tabelle 11-4 e 11-5 mostrano i dati di utilizzo.

LINE DRIVER 8T15:

TABELLA 11-4

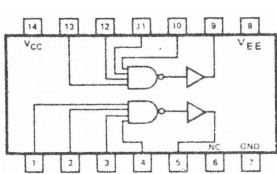
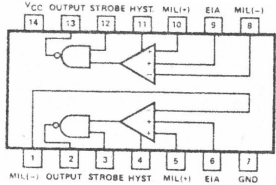
SPECIFICATION	EIA RS-232B, C	MIL-STD-188B	CCITT V24	SIGNETICS 8T15	
				(LIMIT)	(TYPICAL)
Output Voltage "1"	-5Vmin.(R_L = 3000Ω) -15Vmax.(R_L = 7000Ω)	-6 ±1V	-5Vmin. (R_L = 3000Ω) -15Vmax.(R_L = 7000Ω)	-5Vmin. -7Vmax at 4mA	-6V at 4mA
Output Voltage "0"	+5Vmin.(R_L = 3000Ω) +15Vmax.(R_L = 7000Ω)	+6 ±1V	+5Vmin.(R_L = 3000Ω) +15Vmax.(R_L = 7000Ω)	+5Vmin., 7Vmax. at -4mA	+6V at -4mA
Source Impedance (power on)	Not Specified	100Ω max. for I < 10mA	Not Specified		95Ω for ±(0.5 to 4.0 mA)
Source Impedance (power off)	300min. at ±2V	N/A	300min. at ±2V	300min. at ±2V	2 5MΩ
Max. Short Circuit Current	±500mA max. (to ±25V)	100mA max. (to ground)	±500mA max. (to ±25V)	±25mA max. (to ±25V)	±5mA (to ±25V)
Wave Shape (rise and fall time)	±4% of pulse Interval (max.)	±5% of pulse Interval (min.)		4μs-3000pF 200ns-20pF	2μs-3000pF 25ns-20pF
Bit Rate	0-20KHz	4KHz normal	20KHz max.		3MHz
Open Circuit Drive	±25V max.	±6V ±1V		±6V ±1V	±6V
Signal Characteristics	1ms max. transition		1ms max. transition		2μs with C_L = 3000pF
	30V/μs max. dV/dt		30V/μs max. dV/dt		20V/μs with C_L = 500pF

LINE RECEIVER 8T16:

TABELLA 11-5					
SPECIFICATION	EIA RS-232B, C	MIL-STD 188B	CCITT V24	SIGNETICS 8T16	
				LIMIT	TYPICAL
Input Thresholds (V) Max.	+3V, -3V	Not Specified	+3V, -3V	+3V, -3V (1) +0.9V, -0.9V (2) +3V, +0.3V (3)	+2V, -2V (1) +0.6V, -0.6V (2) +2.1V, +1V (3)
Input Thresholds (1) Max.	Not Specified	0.1mA max.	Not Specified	.1mA max.	0.050mA
Input Resistance	3K min., 7K max.	6K min.	3KΩ min., 7K max.	3K min. 7K max. (EIA) 7.5K min. (MIL)	5K (EIA) 12K (MIL)
Hysteresis	Not Specified	Not Specified	Not Specified	2.4V min. (EIA) 0.7V min (MIL)	4V (EIA) 1.2V (MIL)
Max. Input Voltage	±25V (min.)	Not Specified	±25V (min.)	±25V (EIA AND MIL)	
V_{CC}	Not Specified	Not Specified	Not Specified	+7V	+5V ±5%

NOTES: 1) EIA Input, Hysteresis terminal open 2) MIL Input 3) EIA Input, Hysteresis terminal grounded

DISPOSIZIONI DEI PIN

Le disposizioni dei pin dei chip 8T15 e 8T16 sono indicate nei due schemi riportati sotto:

8T15

8T16

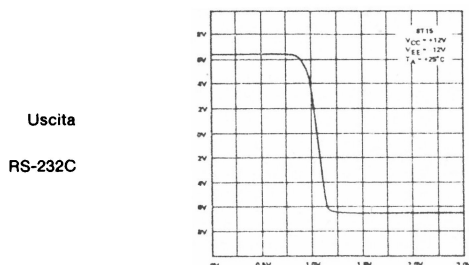
Per il Line Driver 8T15, la Vcc è di + 12 o di + 15 Volt e la VEE è di —12 o —15 Volt rispettivamente. Per il Line Receiver invece la Vcc è normalmente di + 5 Volt.

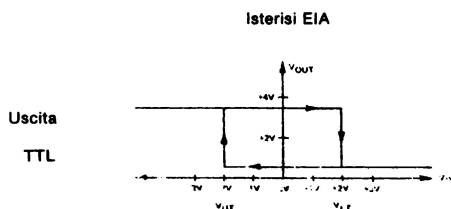
Ciascun driver di un 8T15, svolge una funzione logica NAND a 4 ingressi e può accettare i livelli logici standard TTL. L'uscita presenta un buffer per pilotare linee di interfaccia con livelli di valori normali di + 6 volt e —6 volt. Le uscite sono protette contro eventuali danni provocati da picchi di tensione di ± 25 volt. Il driver dal pin 1 al 5 è utilizzato come interfaccia da TTL a RS - 232C. Il driver dal pin 9 al 13 è disponibile per altre applicazioni. 5 jack non saldati di interconcessione BP25, vengono utilizzati come ingressi e uscite al secondo line driver.

Il Line Driver 8T16 accetta segnali "single-ended" (MIL e EIA) o differenziali (MIL) e li converte ai livelli logici standard TTL. Il circuito si serve di un'isteresi per ottenere alti margini di rumore richiesto quando i ricevitori devono operare in ambienti ad alta densità

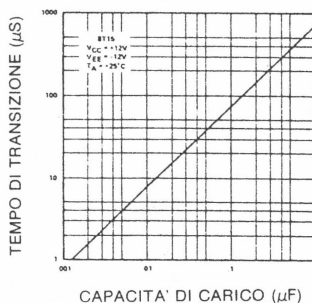
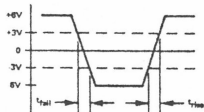
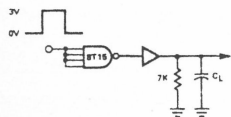
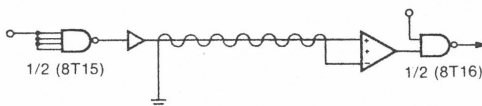
di rumore. Il ricevitore dal pin 9 al 13 è utilizzato come un'interfaccia da RS-232C a TTL. Il ricevitore dal pin 1 al 6 è disponibile per altre applicazioni. 6 jack non saldati, di interconnessione BP25, vengono utilizzati come ingressi e uscite al secondo line receiver. E' importante notare che quando si usa l'ingresso EIA, entrambi gli ingressi MIL devono essere a massa. L'ingresso EIA può essere lasciato aperto se inutilizzato. L'ingresso di strobe opera come segue:

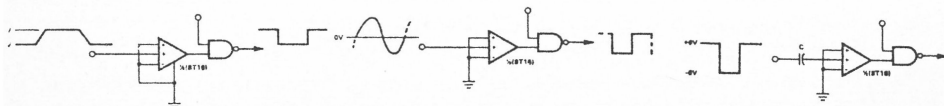
- Uno stato logico 1 (logica positiva) sull'ingresso di strobe permette di trasferire i dati.
- Uno stato logico 0 (logica positiva) sull'ingresso di strobe mantiene un'uscita alta.

Un curva tipica di trasferimento del line driver è mostrata di seguito. Eccetto la variazione di $\pm$ 6 volt della tensione di uscita, la curva è essenzialmente identica a quella della porta standard TTL.

Ingresso TTL

Le caratteristiche di una tipica isteresi dell'8T16, usato come line receiver RS-232C sono mostrate di seguito.

Ingresso RS-232C

APPLICAZIONI TIPICHE PER L'OUTBOARD® LR-13

Di seguito sono mostrate alcune applicazioni suggerite dalla Signetics per i chip 8T15/16. L'ampiezza e la frequenza dell'impulso dei dati dell'8T15, possono essere variate aggiungendo un condensatore tra l'uscita dell'8T15 e massa. Il tempo di transizione diventa quindi una funzione lineare della capacità di carico, come mostrato nel grafico seguente.

La combinazione driver/receiver può operare sia con un alto differenziale, o altrimenti con un alto modo comune di immunità al rumore.

Applicazioni tipiche dell'8T16 sono mostrate nelle seguenti figure.

TRIGGER DI SCHMITT **CONVERTITORE DI ONDE SINUSOIDALE** **OPERAZIONE "AC COUPLED"**

INTRODUZIONE AGLI ESPERIMENTI

In questo capitolo si dovranno collegare una varietà di circuiti che permetteranno al Trasmettitore/Ricevitore Universale Asincrono (UART) di comunicare sia con se stesso che con una telescrivente o altri dispositivi di telecomunicazione che accettano parole di dati di 20 mA current loop. Con questi esperimenti si intende mostrare come l'uso di un UART è di un Outboard* di interfaccia TTL/20 mA possa semplificare un sistema di interfacciamento tra i circuiti digitali ed i comuni dispositivi di ingresso/uscita.
Gli esperimenti di questo Capitolo sono riassunti come segue:

Esperimento N. 1 Provare a trasmettere una parola binaria di 8 bit, in modo seriale, dal trasmettitore al ricevitore di uno stesso UART.

Esperimento N. 2 Esaminare il comportamento delle uscite ai pin 19, 22, 24 e 25 di un chip UART.

Esperimento N. 3 Dimostrare come si possano controllare i bit che costituiscono il carattere trasmesso in modo asincrono da un UART.

Esperimento N. 4 Provare la trasmissione full duplex di un carattere in modo seriale asincrono tra due UART indipendenti.

Esperimento N. 6 e Esperimento N. 7 Provare le funzioni di un chip UART, usato come ricevitore, ad una velocità di trasmissione di 110 baud, collegato ad una telescrivente. E' necessario per questi esperimenti un Outboard* di interfaccia LR-14 TTL/20 mA current loop.

Esperimento N. 8 Provare il funzionamento di un UART usato come un trasmettitore, collegato ad una telescrivente.

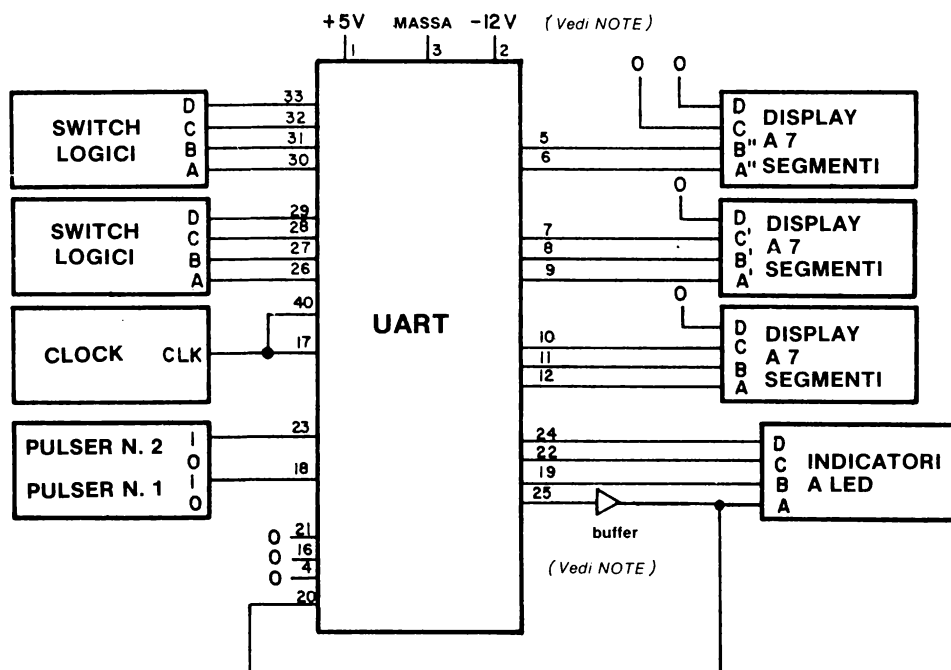
Esperimento N. 9 Provare la trasmissione half-duplex di un carattere ASCII di 11 bit, in modo seriale asincrono, tra una telescrivente ed un chip UART.

Esperimento N. 10 Provare la trasmissione full-duplex di un carattere ASCII di 11 bit, in modo seriale asincrono, tra una telescrivente ed un chip UART.

ESPERIMENTO N. 1

Scopo

Questo esperimento consente di provare ad inviare dal trasmettitore al ricevitore di uno stesso UART una parola binaria di 8 bit in modo seriale. *Utilizzate il seguente circuito anche per l'Esperimento N. 2.*

Schema del circuito

NOTE: 1. Per l'UART C MOS IM 6402, il pin 2 non è usato.
2. Il buffer può essere un chip 7408 oppure una coppia di inverter 7404.
3. Per l'UART C MOS IM 6403 i pin 2,17 e 40 hanno delle funzioni speciali.

Passo 1

Studiate in modo accurato lo schema mostrato sopra. Prima di collegare il circuito tenete presente i seguenti suggerimenti e precauzioni.

- Maneggiate il cavo Augat a 16 fili con estrema attenzione. I pin dei connettori sono molto fragili e si possono rompere con estrema facilità. Proteggete i pin dei connettori quando il cavo non è usato. Per agevolarvi vengono fornite protezioni in plastica Styrofoam.

- Se usate un chip UART C MOS, l'alimentazione C MOS deve essere +5 volt, al fine di essere TTL compatibile sia con gli ingressi che con le uscite. Se usate una batteria di 6 volt per i chip TTL, usate la stessa batteria per alimentare il chip UART C MOS.
- Se usate un UART C MOS, i pin inutilizzati non devono essere lasciati disconnessi. Gli ingressi a 0 logico devono essere posti a massa. Gli ingressi ad 1 logico devono essere collegati a Vcc tramite un resistore pull-up da 10 kΩ.
- L'UART C MOS richiede una sola alimentazione con tensione da + 4 a + 7.
- Il chip UART standard è un circuito integrato MOS LSI (Metal-Oxide-Semiconductor). Le uscite da questo chip hanno delle caratteristiche di fan-out molto basse, il che significa semplicemente che sarete in grado di collegare una uscita ad un chip con un fan-in di 0,1. Nello schema, potete osservare che, tra l'uscita del pin 25 e l'indicatore a LED A, è posto un buffer. Il Fan-out del pin 25 è 1 carico TTL, quindi dovete usare un buffer, tipo la porta AND 7408, al fine di aumentare il fan-out. Se eliminate il buffer, potreste non vedere alcuna lettura, a parte 000, in codice ottale, sui 3 display a led a 7 segmenti. Noi abbiamo incontrato questo problema quando collegammo per la prima volta il circuito, e ci servirono parecchie ore per eliminare l'inconveniente.
- L'uscita dell'UART appare in codice ottale sui tre display a 7 segmenti. Consultate il Capitolo 5 del Bugbook I per avere maggiori dettagli sul significato di "codice ottale".
- Quando utilizzate il cavo AUGAT a 16 fili, controllate che i pin, dal 26 al 33 dell'UART, siano connessi agli switch logici. E' molto facile connettere i cavi in modo inverso.
- Se utilizzate delle batterie per ottenere una tensione di 12 volt, controllate che siano collegate in serie, come in fig. 11-6.
- Questi esperimenti possono essere eseguiti tutti utilizzando la serie IM 6402 dei chip UART C MOS. Se preferite utilizzare la serie IM 6403, fate attenzione alle funzioni speciali dei pin 2,17 e 40 ed ignorate l'Esperimento 3.
- Se utilizzate l'Outboard LR-21, utilizzate gli otto mini-switch logici oppure del filo N. 24 per i pin 4 e 16. *L'Outboard standard LR-21 può essere utilizzato con un UART C MOS, tipo l'IM 6402. Tuttavia un resistore pull-up da 10 kΩ* deve essere collocato tra ciascuno dei seguenti pin e Vcc: 4, 16, 34, 36, 37, 38 e 39. Vi raccomandiamo altresì di rimuovere il diodo tra il pin di - 12 volt ed il pin 2 dell'UART, per prevenire la possibilità di applicare -12 volt al chip C MOS UART. Se usate l'IM 6403, fate attenzione alle differenti funzioni dei pin 2, 17, 40.

Passo 2

Collegate il circuito come mostrato: tutte le volte che lavorate con l'UART, dovete essere certi che l'alimentatore sia spento quando state collegando un circuito o cambiando le connessioni. E' molto facile far saltare un circuito integrato MOS, piuttosto che un chip TTL della serie 7400.

Consultate la figura 11-9 per sistemare i nove Outboard ed il chip 7404. Utilizzate un cavo Augat a 16 fili per connettere il plug di dati sull'Outboard UART gli switch logici ed i display a 7 segmenti a pag. 11-E6.

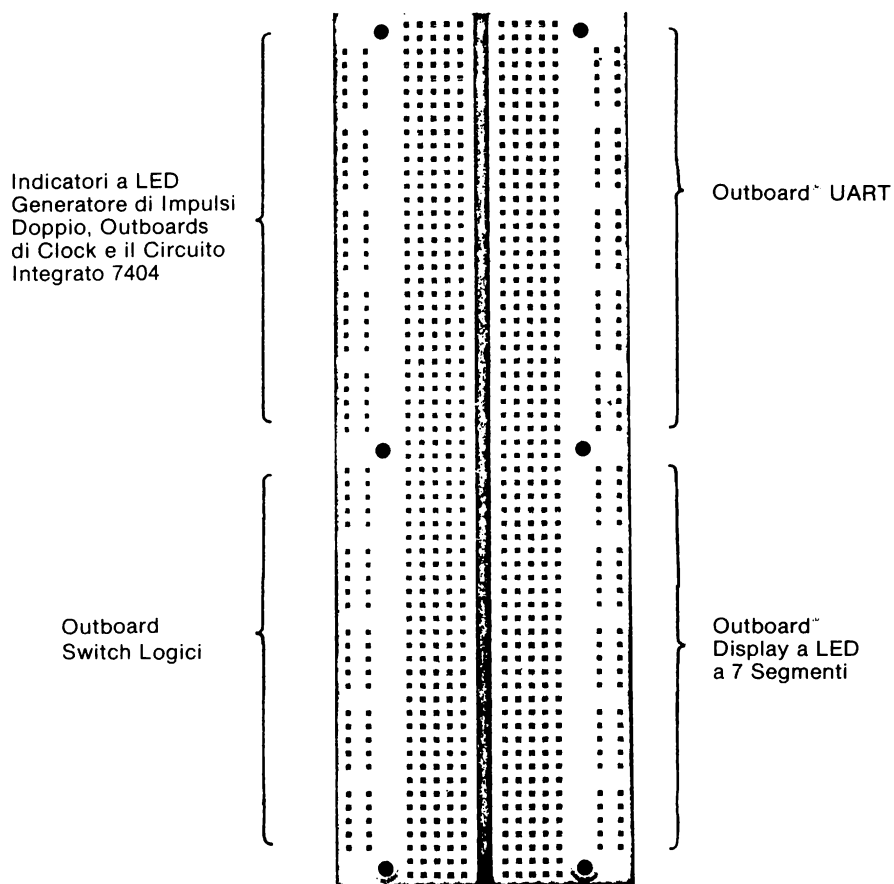



Figura 11-9. Sistemazione dei 9 Outboard e del circuito integrato 7404 nell'Esperimento N. 1. Se i tre display a sette segmenti non stanno nella piastra vista sopra, utilizzate una seconda piastra ed estendete il circuito. Vi consigliamo di porre sulla seconda piastra SK-10 gli switch logici e i tre display a 7 segmenti.

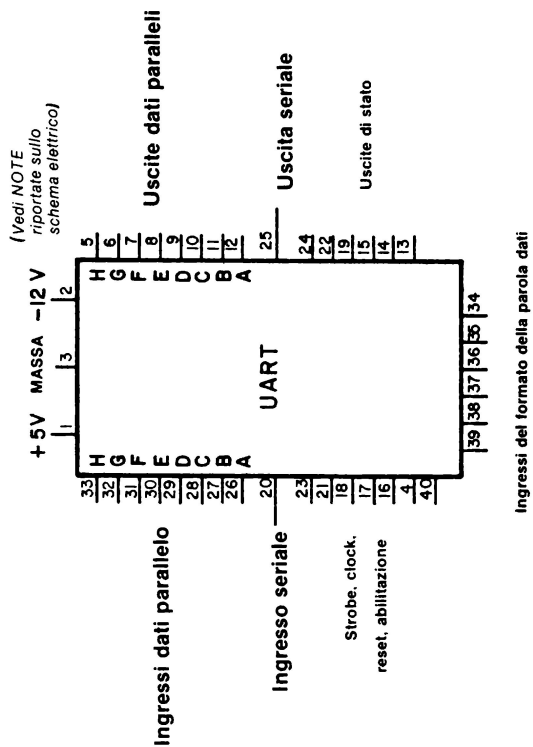



Figura 11-10. Schema a blocchi del chip UART.

Figura 11-11. Fotografia di una tipica connesione del circuito per l'Esperimento N.1

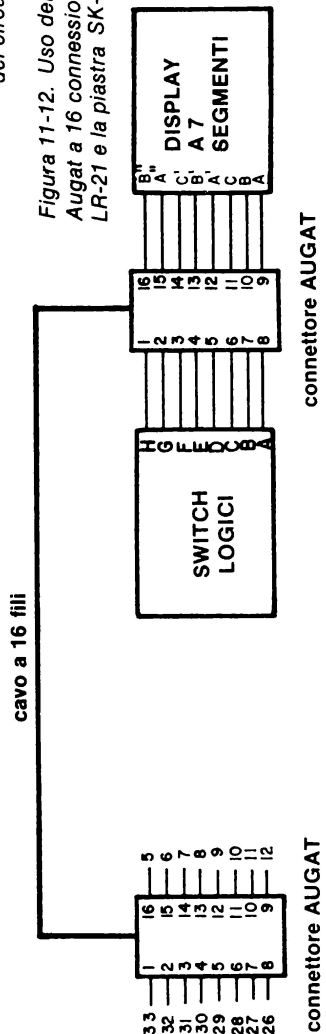
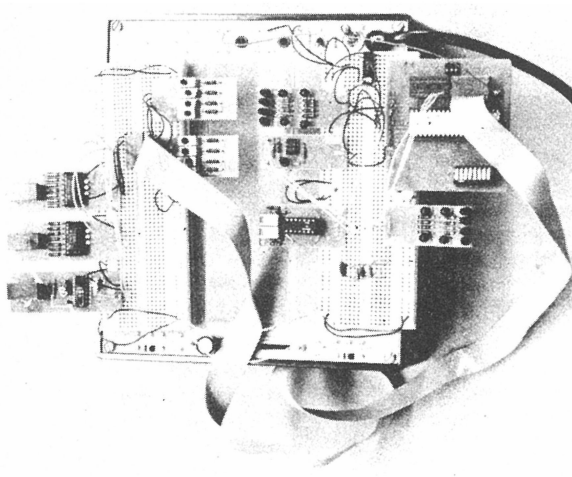



Figura 11-12. Uso del connettore Augat a 16 connessioni tra l'Outboard· LR-21 e la piastra SK-10.

sulla piastra SK-10. Saranno necessari al massimo 11 connessioni tra il connettore Augat e gli Outboard sulla piastra SK-10.

Passo 3

Posizionate gli switch logici da A a H a HGFEDCBA = 00000001. Utilizzate un condensatore di 0,05 o 0,10 μF con un Outboard di clock per produrre una frequenza di clock tra 7 e 15 Hz. Applicate tensione al breadboard. Premete e rilasciate il pulsante del generatore di impulsi N. 1, cioé il RESET DATA AVAILABLE. Non dovrebbe accadere nulla: ora premete e rilasciate il generatore di impulsi N. 2, il DATA STROBE. I quattro indicatori a LED dovrebbero essere OFF per un breve periodo di tempo. L'indicatore a LED C poi si accende e rimane acceso. Gli indicatori a LED B e C si accendono e i tre display a sette segmenti visualizzano 001. Se avete osservato tutto questo, significa che è stata trasmessa in successione una parola di 8 bit dagli switch logici agli indicatori a LED.

Passo 4

Per avere una configurazione degli switch logici di HGFEDCBA = 01001001 dovete premere e rilasciare il generatore di impulsi N. 1 e quello N. 2. Dopo alcuni secondi, tutti i 4 indicatori a LED si accenderanno e sui tre display si potrà leggere 111_8 in codice ottale.

Passo 5

Per finire questo esperimento, sostituite il condensatore di 0,05 e 0,10 μF con uno che abbia una frequenza 50 o 100 volte più piccola. Noi stiamo utilizzando ora un condensatore di 0,001 μF con il nostro Outboard di clock per produrre una frequenza di clock di circa 650 Hz. Con una tale frequenza, occorrono meno di un secondo per trasmettere il set di parole di 8 bit dagli switch logici.

Si confermi o si riempia la parte destra della tabella seguente. Si posizionino gli 8 switch logici in modo appropriato e poi si prema e si rilasci il generatore di impulsi N. 2. Non è necessario resettare l'uscita per ottenere i dati ogni volta, cioé non è necessario premere il generatore di impulsi N. 1 ogni volta.

Valori degli switch logici								*Visualizzazione in ottale sui display a sette segmenti*
H	*G*	*F*	*E*	*D*	*C*	*B*	*A*	
0	0	0	0	0	0	0	1	001
0	1	0	0	1	0	0	1	111
1	0	0	1	0	0	1	0	222
1	1	0	1	1	0	1	1	333
1	1	1	1	1	1	1	1	377
1	0	1	1	0	0	0	0	260
0	0	1	1	0	0	0	0	060
0	1	0	1	0	1	0	1	
1	0	1	0	1	0	1	0	
1	1	1	0	0	0	1	1	
1	1	1	1	1	0	0	0	
1	1	1	1	1	1	0	0	

ESPERIMENTO N. 2

Scopo

Questo esperimento esamina il comportamento delle uscite ai pin 19, 22, 24, e 25 del chip UART. *Utilizzate questo circuito anche per l'Esperimento N. 3.*

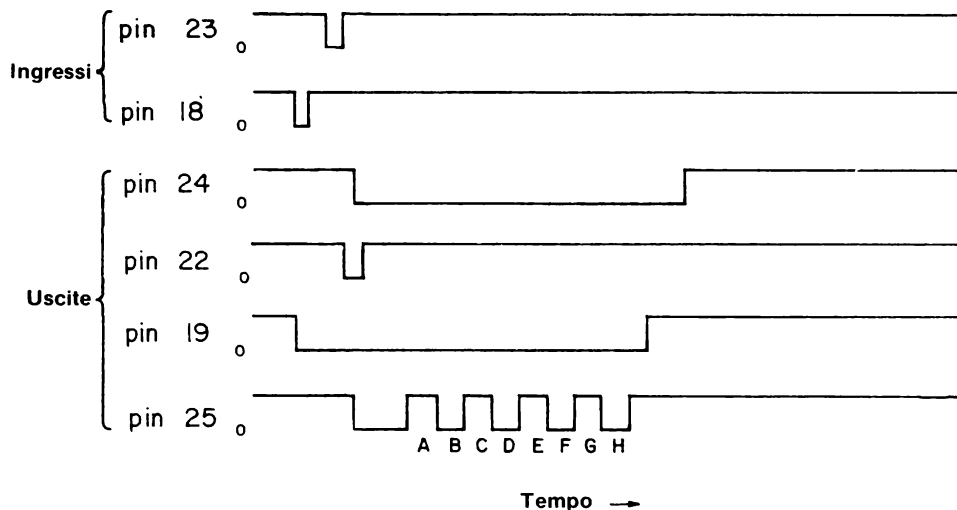
Schema del circuito

E' identico a quello dato nell'Esperimento N. 1.

Passo 1

Sostituite il condensatore sull'Outboard di clock con uno di 0,22 o 0,50 μF, preferibilmente si scelga l'ultimo valore. Un tale condensatore ridurrà la frequenza di clock a 1,5 o 3 Hz che è sufficientemente bassa per osservare l'uscita in modo seriale, al pin 25 e lo stato delle uscite ai pin 19, 22, e 24.

Passo 2

Studiate le forme d'onda date di seguito:

Notate che:

- Un impulso negativo di clock al pin 18 porta lo stato logico dell'uscita dal pin 19 a 0. Questo pin rimane allo 0 logico fino al momento in cui il bit finale della parola di 8 bit si presenta al pin 25 dell'uscita seriale.
- Un impulso negativo di clock al pin 23 porta lo stato logico dei pin di uscita 22, 25 e 24, nell'ordine dato, a 0. Il pin 22 ritorna quasi immediatamente allo stato logico 1 e vi rimane durante l'intera trasmissione della parola di 8 bit.

- Il pin 19 ritorna allo stato logico 1 alla fine della trasmissione del bit di stop finale, in un carattere di 11 bit che contiene un unico bit di start, 8 bit di dati e due bit di stop.
- La parola che è stata trasmessa è HGFEDCBA = 01010101
- Il bit meno significativo della parola, il bit A, è trasmesso per primo e il più significativo, il bit H, per ultimo.

Passo 3

Applicate tensione al breadboard. Ponete gli switch logici a HGFEDCBA = 01010101. Se siete certi che il clock non è più veloce di 1 Hz, premete e rilasciate il pulsante del generatore di impulsi N. 1. L'indicatore a LED B dovrebbe spegnersi sul fronte principale negativo dell'impulso di clock. Probabilmente non riuscirete ad osservare quanto detto perché l'indicatore a LED non era acceso la prima volta. Osserverete il comportamento precedente del pin 19 soltanto quando verrà trasmessa la seconda parola di 8 bit.

Passo 4

Preparatevi ora a premere e rilasciare il generatore di impulsi N. 2. Osservate i tre rimanenti indicatori a LED. Hanno la tendenza a spegnersi nel seguente ordine: C, seguito da A, seguito da D. Premete e rilasciate il generatore di impulsi ed osservate se ciò avviene. *Osservate in modo continuato, per uno o due minuti successivi, tutti e 3 gli indicatori a LED e scrivete nello spazio seguente cosa accade:*

Abbiamo osservato che il pin 19 si posiziona allo stato logico 1, esattamente otto impulsi di clock dopo l'inizio della trasmissione del primo bit di stop (dopo l'MSB della parola). Inoltre il pin 24 si posiziona allo stato logico 1, esattamente 32 impulsi di clock dopo l'inizio della trasmissione del primo bit di stop. Chiaramente, il pin 24 si posiziona allo stato logico 1 solo alla fine della trasmissione del carattere di 11 bit.

Passo 5

Ripetete il passo 4 ed osservate che il pin 19 si posiziona allo stato logico 0 appena si preme il generatore di impulsi N. 1. Confermate le vostre osservazioni del passo 4.

Ripetete il passo 4 volte quante sono necessarie per trovare le stesse forme d'onda mostrate nel passo 2.

Discussione

Il pin 19 è il pin d'uscita di DATA AVAILABLE sul chip UART. Secondo le specifiche della General Instruments, questa linea si mette allo stato logico 1 quando un'intera parola è stata ricevuta e trasferita al registro del ricevitore. L'uscita, nel vostro caso, si è portata allo stato logico 1 solo quando il bit dei dati H, l'ultimo bit dei dati, è stato trasmesso.

Il pin 25 è il pin di SERIAL OUTPUT. Questa linea di uscita trasmetterà, bit dopo bit, in modo seriale l'intero carattere da trasmettere, in questo caso HGFEDCBA = 01010101. Questa linea rimane allo stato logico 1 quando non vengono trasmessi alcuni dati. Questo comportamento è confermato nella serie di forme d'onda mostrate prima.

Il pin 24 è il pin di uscita di END OF CHARACTER. Questa linea di uscita si porta allo stato logico 1 ogni volta che un carattere intero viene trasmesso. Per "carattere intero" si intende il bit di start, i bit dei dati, il bit di parità e uno o due bit di stop. La linea rimane allo stato logico 1 finché non viene trasmesso il successivo carattere posizionandosi allo stato logico 0.

Il pin 22 è il pin di uscita TRASMITTER BUFFER EMPTY. Questa linea di uscita è all'1 logico quando il registro, che contiene i bit dei dati, può essere caricato con un altro carattere.
Come è mostrato nella serie di forme d'onda digitali, si ha la possibilità di caricare un'ulteriore parola nel registro di buffer quando si decide di farlo.

Tre delle precedenti uscite - i pin 19, 22 e 24 - possono essere considerate come uscite di stato. Forniscono le informazioni che sono necessarie al chip UART. Tali informazioni possono essere fornite ed altre parti di un circuito digitale complesso, che incorpora il chip UART. Tali uscite rendono il chip UART potente e molto flessibile.

Nelle specifiche del costruttore, si può osservare che i simboli per i pin 18 e 23 sono soprassegnati. Per esempio, nelle specifiche della General Instruments per il chip UART AY-5-1013/AY-5-1013A, i simboli per questi due pin sono:

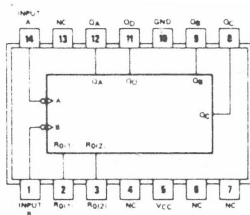
Pin	*Simboli*
18	$\overline{RDAV}$
23	$\overline{DS}$

La segnatura indica che questi pin di ingresso devono essere allo stato logico 1 quando non svolgono nessuna funzione. Un breve impulso di clock negativo abiliterà ciascun pin a svolgere le proprie funzioni.

ESPERIMENTO N. 3

Scopo

Questo esperimento dimostra come si può controllare il numero di bit nel carattere trasmesso in modo asincrono e seriale da un UART.

Configurazione dei pin del circuito integrato

7493

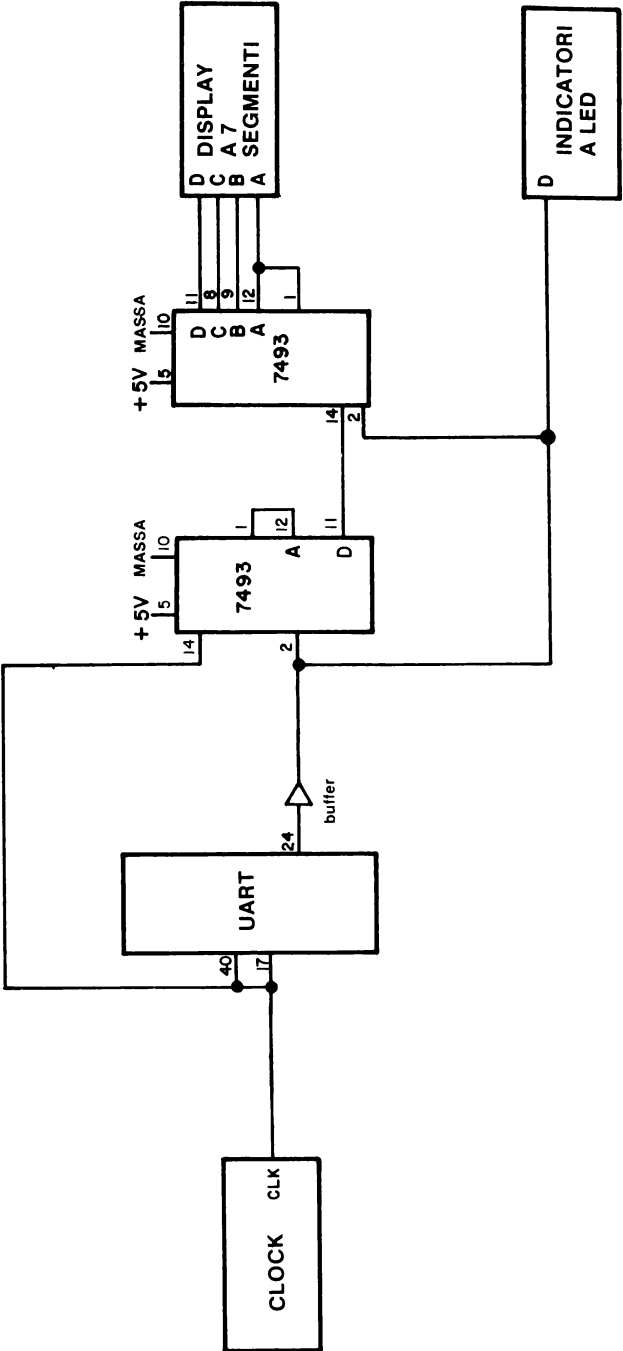
Schema del circuito

Molti di questi circuiti sono identici a quello dato nello schema elettrico dell'Esperimento N. 1. Alcune aggiunte o modifiche che possono essere fatte sono mostrate nello schema della pagina seguente.

Passo 1

Studiate le modifiche e le aggiunte del circuito costruito negli Esperimenti N. 1 e 2. E' necessario un'ulteriore piastra SK-10. E' necessario che sia sottoposto a buffer, per incrementare il suo fan-out. Si raccomanda l'uso di un Outboard LR-12 Driver/Inverter/NOR o un paio di inverter 7404.

Il circuito integrato 74193 è usato come contatore "divisore per 16".

CLOCK
CLK
UART
buffer
+5V
MASSA
7493
A
D
+5V
MASSA
7493
D C B A
DISPLAY
A 7
SEGMENTI
D C B A
INDICATORI
A LED
D

Il pin 12 dell'uscita di carry pilota un contatore binario 7493 a 4 bit, che a sua volta pilota un display a LED a 7 segmenti.

Per un carattere di 7 bit asincrono seriale, appariranno i seguenti numeri sul display prima che sia resettato a 0 decimale,

0, 1, 2, 3, 4, 5, 6

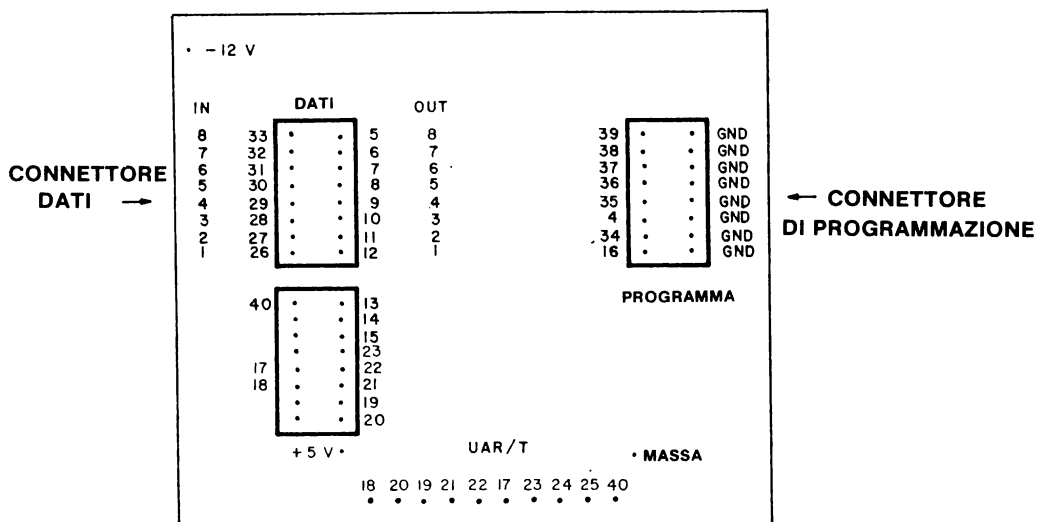
Si può osservare che ci sono 7 bit anche se il conteggio termina al 6 decimale. Si tenga a mente come si può variare la lunghezza del carattere asincrono seriale; il numero totale di bit nel carattere sarà sempre uno in più del numero più grande, o del simbolo che appare sul display a LED a 7 segmenti.

Passo 2

Fate le modifiche al circuito dell'Esperimento N. 1 come indicato nello schema precedente. *E' necessario un ulteriore piastra SK-10 per questo Esperimento!* Noi abbiamo disposto i contatori 7493 e 74193 sulla seconda piastra.

Sistemate la frequenza di clock in modo che sia approssimativamente da 10 a 30 Hz. Noi abbiamo utilizzato un condensatore di 0,05 microfarad con il nostro Outboard di clock ottenendo una frequenza di clock di 15 Hz.

Passo 3

Applicate tensione al breadboard e posizionate gli switch logici a HGFEDCBA = 01010101. I cambiamenti che dovrete fare, saranno localizzati sul connettore di programmazione dell'Outboard UART, come mostra la seguente figura:

Passo 4

I pin dal 35 al 39 controllano importanti caratteristiche del carattere seriale asincrono, che includono: il numero dei bit di stop, il numero dei bit di dati, la presenza o l'assenza di un bit di parità e l'indicazione del tipo di parità, se il bit di parità è pari o dispari. Il pin 34 deve rimanere allo stato logico 1 durante questo esperimento.

Utilizzate soltanto il filo N. 24 nel connettore di programmazione sull'Outboard UART. Le caratteristiche della piastra non consentono di utilizzare il filo N. 22.

Per il vostro primo test della possibilità di programmazione dell'UART, non collegate a Massa (GROUND) nessun filo dei pin dal 35 al 39. Lasciate tutti i 5 pin allo stato logico 1. Premete e rilasciate il generatore di impulsi N. 2 in modo da trasmettere un unico carattere asincrono e seriale dallo switch logico collegato al trasmettitore, ai display collegati al ricevitore. [In questo Esperimento, non è importante conoscere lo stato logico dei tre bit meno significativi della parola; conseguentemente, si può togliere il display a 7 segmenti utilizzato per questi 3 bit e collegarlo all'uscita del contatore 7493]. Qual'è il numero decimale o il simbolo più grande che osservate sul display collegato al contatore 7493? Scrivete la risposta di seguito.

Passo 5

Se avete svolto correttamente l'esperimento dovreste osservare il simbolo ⊏, che corrisponde al 10 decimale sul display. Il display dovrebbe produrre una sequenza da 0 a ⊏ e pòi ritornare a 0,

0, 1, 2, 3, 4, 5, 6, 7, 8, 9, ⊏ , 0

Una tale sequenza corrisponde a *11 bit* nel carattere seriale asincrono.

Nello spazio seguente, scrivete il numero di bit di start, bit di dati, bit di parità e bit di stop del carattere precedente di 11 bit:

bit di start

bit di dati

bit di parità

bit di stop

Se non siete in grado di rispondere, procedete con l'esperimento e ritornate a questo problema quando pensate di poter rispondere. [*SUGGERIMENTO*: leggete le specifiche del costruttore dell'UART. Determinate ciò che dice il costruttore circa gli ingressi ai pin dal 35 al 39].

Passo 6

Collegate ora il pin 35, il pin di ingresso di NO PARITY, Massa (GROUND) del connettore di programmazione e premete e rilasciate il generatore di impulsi N. 2 determinando il numero di bit del nuovo carattere asincrono seriale che si è creato. Scrivete il numero di bit nello spazio seguente.

Avrete osservato che ci sono ora *12 bit* nel carattere asincrono seriale. Avete aggiunto un bit di parità tra l'ultimo bit dei dati e il primo bit di stop! Quando il pin 35 è allo stato logico 1, non c'è il bit di parità, quando invece è allo stato logico 0, il bit di parità esistente avrà uno stato logico dato da quello del pin 39 a seconda che la parità sia pari o dispari.

Passo 7

Ora ponete a massa i pin 37 e 38, premete e rilasciate il generatore di impulsi N. 2. Scrivete di seguito il numero di bit relativi al carattere trasmesso.

Avrete osservato che ora sono presenti soltanto *9 bit*. Riferendosi alle specifiche del costruttore dell'UART scrivete qui di seguito il numero dei bit di start, dei bit di dati, dei bit di parità e dei bit di stop che costituiscono il carattere di 9 bit asincrono seriale.

bit di start

bit di dati

bit di parità

bit di stop

Se non sapete rispondere a queste domande procedete pure con l'esperimento.

Passo 8

Fate ora ritornare il pin 35 allo stato logico 1. Facendo questo si elimina il bit di parità. In aggiunta ponete il pin 36, il pin di ingresso di NUMBER OF STOP BIT, a Massa. Premete e rilasciate il generatore di impulsi N. 2 e scrivete nello spazio seguente, il numero di bit del *carattere più piccolo che può essere trasmesso dall'UART.*

A questo punto non vi daremo alcuna risposta. Dovreste ormai essere in grado di determinare il numero di bit senza difficoltà.

Quanti bit di dati ci sono nel carattere più piccolo? Quanti bit di stop? Quanti bit di start? Quanti bit di parità?

I pin dal 35 al 38 controllano il numero di bit della parola asincrona seriale trasmessa dall'UART. Ad eccezione dei pin 13, 14, 15 e 39 avete ora provato o collegato tutti i pin del chip UART a 40 pin! In questo modo avete visto tutte le funzioni di questo chip. Se non le avete comprese ripetete gli Esperimenti dal N. 1 al N. 3.

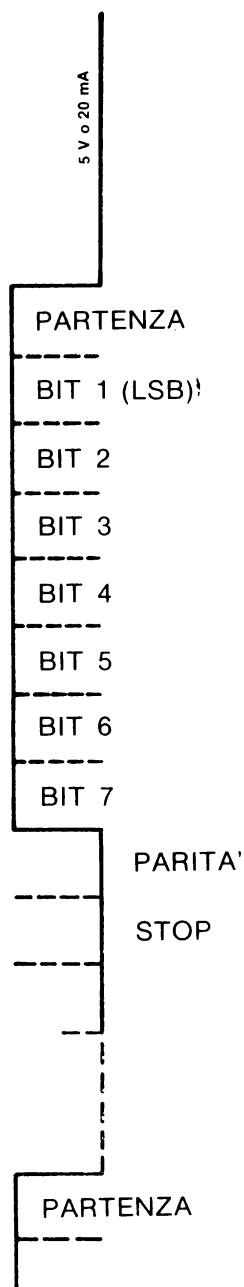



Figura 11-13. Una tipica sequenza di bit di un carattere ASCII asincrono seriale a 11 bit. Notate che il bit di start è allo 0 logico e i bit di stop sono all'1 logico.

Passo 9

Esaminerete ora il comportamento del bit di parità e la funzione di ingresso di ODD/EVEN PARITY SELECT del pin 39.

Selezionate il pin 35 allo stato logico 0; in questo modo è stato aggiunto il bit di parità subito dopo i bit del carattere asincrono seriale. Selezionate inoltre il pin 38 allo stato logico 0; la parola dei dati in questo modo è costituita da 7 bit. Il bit ottavo, che corrisponde allo switch logico H, è stato tolto. Questa situazione è mostrata in figura 11-13, che rappresenta un tipico carattere ASCII seriale asincrono di 11 bit, costituito da un unico bit di start, 7 bit di dati, un bit di parità e due bit di stop. Notate che il bit 0 è il bit di start., il bit 1 è l'LSB, il bit 7 è l'MSB, il bit 8 è il bit di parità ed infine i bit 9 e 10 sono i bit di stop.

Se il pin 39 è allo stato logico 1, un bit può, oppure non può, essere inserito nella posizione del bit di parità cosicché il carattere seriale asincrono avrà una *parità pari*. Ricordate che la definizione di "parità" è:

parità — Un metodo per determinare che non vi siano errori nel numero binario. Un bit extra, chiamato bit di parità, è aggiunto ai bit del carattere. Se il bit di parità è utilizzato, la somma di tutti gli "1" nel numero binario e il suo bit di parità corrispondente è sempre pari. Se è utilizzata la parità dispari, la somma degli "1" e il bit di parità è sempre dispari.

Ci si può chiedere se questo bit di parità è utile e inoltre se i bit di stop sono inclusi oppure no dalla somma della parità.

Passo 10

Quando il pin 36 è allo stato logico 1, si hanno due bit di stop, entrambi allo stato logico 1. In questo modo non si può determinare se i bit di stop sono o non sono inclusi nella somma della parità. Ponete il pin 36 allo stato logico 0. Vi sono ora 10 bit nel carattere asincrono.

Ponete gli switch GFEDCBA = 0000000 e determinate lo stato logico del bit di parità, che è il bit 8 sul display a LED a 7 segmenti connesso al contatore binario 7493. Dovete regolare la frequenza di clock in modo da vedere lo stato logico del bit 8. Scrivete di seguito lo stato logico.

Noi abbiamo osservato uno stato logico 0 al bit seriale numero 8. Poiché il pin 39 è all'1 logico e abbiamo una parità pari, dobbiamo concludere che il bit di stop non è incluso nella somma di parità. Ciò è comprensibile poiché il bit di stop è in sequenza, subito dopo l'assegnamento della parità, al bit numero 8.

Passo 11

Ponete il pin 39 allo 0 logico corrispondente ad una parità dispari. Determinate lo stato logico del bit di parità quando i dati sono così costituiti: GFEDCBA = 0000000.

Abbiamo osservato che lo stato logico del bit numero 8 nella parola asincrona cambia all'1 logico. *Uno stato logico 1 è stato inserito dall'UART, nella posizione del bit di parità, per rendere la parità del carattere dispari!* [Ricordate che il bit numero 8, che appare sul display a LED a 7 segmenti, è attualmente il nono bit nel carattere a 10 bit].

Passo 12

Nella tabella seguente inserite sia i valori previsti che quelli osservati, per lo stato logico del bit di parità in un carattere asincrono seriale a 10 bit. Ponete lo stato logico del pin 39 in modo da avere o la parità pari o quella dispari.

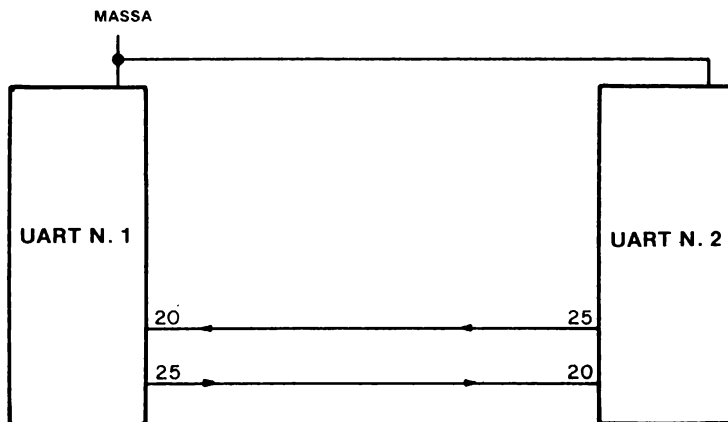
								Stato logico del bit di parità	
G	*F*	*E*	*D*	*C*	*B*	*A*	*Parità desiderata*	*Previsto*	*Osservato*
0	0	0	0	0	0	0	pari	0	0
0	0	0	0	0	0	0	dispari	1	1
0	0	0	0	0	0	1	dispari		
0	0	0	0	0	1	0	dispari		
0	0	0	0	0	1	1	dispari		
0	0	0	0	1	0	0	dispari		
0	0	0	0	1	0	1	dispari		
0	0	0	0	1	1	0	dispari		
0	0	0	0	1	1	1	dispari		
0	0	0	1	0	0	0	dispari		
0	0	0	1	0	0	1	dispari		
0	0	0	0	0	0	0	pari	0	0
0	0	0	0	0	0	1	pari		
0	0	0	0	0	1	0	pari		
0	0	0	0	0	1	1	pari		
0	0	0	0	1	0	0	pari		
0	0	0	0	1	0	1	pari		
0	0	0	0	1	1	0	pari		
0	0	0	0	1	1	1	pari		
0	0	0	1	0	0	0	pari		
0	0	0	1	0	0	1	pari		

Il concetto di parità diventerà più significativo quando si costruiranno due UART per trasmettere dati seriali da uno all'altro.

ESPERIMENTO N. 4

Scopo

Questo esperimento prova la trasmissione in full duplex di un carattere seriale asincrono tra due UART indipendenti. L'osservazione più importante che si può fare è che le *frequenze di clock degli UART non devono variare più del 5% l'una dall'altra.*

Schema del circuito

Passo 1

L'esperimento deve essere fatto da due persone. Ciascuna dovrà avere il modello dei collegamenti dell'Esperimento N. 1 e 2. Bisogna però interconnettere il ricevitore di un UART con un trasmettitore dell'altro e viceversa, come mostra la figura precedente. *Nessun altro cambiamento è necessario rispetto al circuito dell'Esperimento N. 1.*

Passo 2

Con gli UART non alimentati, eseguite i cambiamenti indicati nel circuito. Assicuratevi di avere un'unica massa per i due UART separati. Assicuratevi anche, che ciascun UART abbia la sua connessione all'alimentatore, ricordandosi che lo scopo di questo esperimento è di dimostrare che due UART alimentati indipendentemente possono comunicare l'uno con l'altro, tenendo solo presente che le loro frequenze di clock non varino l'una dall'altra più del 5%.

Passo 3

Determinate la frequenza di ciascuno dei clock degli UART utilizzando un contatore divisore-per sedici mostrato nell'Esperimento N. 3 ed un cronometro. Fate poi in modo che la frequenza sia circa tra i 5 e i 15 Hz. Si può utilizzare a tale scopo una tecnica di misurazione delle frequenze, discussa nel Capitolo 4 del Bugbook I. Differentemente si può usare un frequenzimetro per determinare la frequenza del suo clock.

Nel nostro caso, abbiamo misurato una frequenza di clock di 11,3 Hz all'UART N. 1 e di 10,4 Hz all'altro: di conseguenza tali frequenze variano più del 5% tra loro.

Passo 4

Quando le due frequenze di clock non variano più del 5%, provate ad inviare un carattere seriale asincrono tra i due UART. Date alimentazione ad entrambi gli UART e premete il generatore di impulsi N. 2 per l'UART N. 2 ed osservate se l'informazione è trasmessa oppure no all'UART N. 1. Ripetete la procedura, ma questa volta premendo il generatore di impulsi N. 2, sull'UART N. 1 ed osservate se l'informazione è stata trasmessa oppure no all'UART N. 2. Se le frequenze di clock sono oltre il 5%, non siete in grado di trasmettere caratteri senza errori da un UART ad un altro. Variate le posizioni degli switch logici e dimostate quanto è stato detto.

Nel caso del nostro esperimento, abbiamo trasmesso una parola di dati HGFEDCBA = 01010101 dall'UART N. 2 al N. 1. Eravamo sicuri di aver programmato ciascun UART per trasmettere e ricevere 8 bit di dati, nessun bit di parità e due bit di stop, per un totale di 11 bit con incluso il bit di start. Abbiamo premuto e rilasciato il generatore di impulsi N. 2 sull'UART N. 2. Dopo alcuni secondi abbiamo osservato il numero ottale 256_8 sui tre display a LED a 7 segmenti connessi all'UART N. 1. *Questi dati di uscita erano sbagliati, per il fatto che le frequenze di clock dei due UART erano diverse tra loro di un valore superiore al 5%.*

Abbiamo poi ridotto la frequenza di clock dell'UART N. 1 aggiungendo un condensatore di 0,005 microfarad al condensatore di 0,05 microfarad già presente. La nuova frequenza di clock all'UART N. 1 è diventata quindi 10,2 Hz, un valore diverso da quello dell'UART N. 2, 10,4 Hz ma meno del 5%. Abbiamo trasmesso di nuovo la parola di dati, HGFEDCBA = 01010101 dall'UART N. 2 all'UART N. 1 osservando il codice ottale 125_8 sui 3 display a 7 segmenti connessi all'UART N. 1. *Questi dati di uscita erano corretti, come abbiamo potuto constatare confrontando la parola dei dati originaria.*

Concludiamo, in base agli esperimenti precedenti, che soltanto quando le due frequenze hanno valori che non variano più del 5% tra loro, si possono trasmettere correttamente dati in modo seriale asincrono.

Passo 5

Le frequenze di clock non bisogna farle variare tra loro più del 5%, ed in particolare è raccomandabile una variazione massima del 3% o ancora meno. Ricordate che quanto più grande è la capacità del condensatore sull'Outboard di clock, tanto più piccola è la frequenza di clock. Inoltre la capacità di un condensatore connesso in parallelo si somma a quella precedente.

Quando le due frequenze sono corrette cercate di mandare i dati da un UART all'altro. Per questo bisogna solo premere e rilasciare il generatore di impulsi N. 2 sull'UART che deve trasmettere il carattere. Se volete in seguito ricevere il carattere asincrono, dovete premere e rilasciare il generatore di impulsi N. 1 su l'UART ricevente. Quando questo UART riceve l'intera parola di dati, il pin 19 si posizionerà allo stato logico 1 dallo stato logico 0.

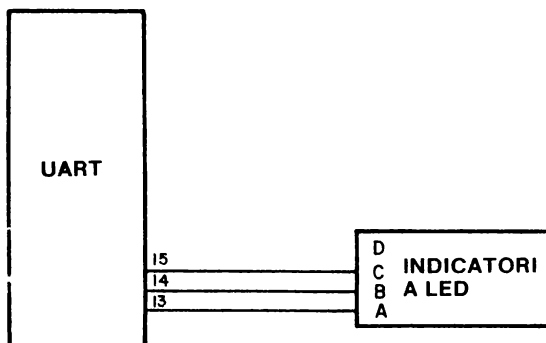
Continuate a trasmettere i dati da un UART all'altro fino a che non vi sarete annoiati. Quando terminate l'esperimento, mantenete gli stessi collegamenti elettrici per il prossimo Esperimento N. 5, dove studierete le caratteristiche dei pin dal 13 al 15 di un UART.

ESPERIMENTO N. 5

Scopo

Questo esperimento dimostra il comportamento dei pin di uscita 13 e 15 del chip UART. In questo esperimento si riutilizza il circuito dell'Esperimento N. 4.

Schema del circuito

Il circuito è essenzialmente lo stesso di quello dell'Esperimento N. 4. I pin 13 e 15 sono connessi all'indicatore a LED, come è mostrato nello schema seguente.

Passo 1

Consigliamo di utilizzare gli indicatori a LED connessi precedentemente ai pin 22, 24 o 25 su uno degli UART. Fate la connessione con l'aiuto di un *filo N. 24*. Tenete presente che i terminali sull'Outboard di UART non possono accettare il filo N. 22.

Passo 2

Dopo aver collegato i due circuiti all'alimentatore mette a massa il pin 35 di ciascun Outboard di UART. In questo modo verrà trasmesso un bit di parità in ciascun carattere asincrono. In più il ricevitore dovrà ora scegliere la parità del carattere.

Passo 3

Collegate a massa il pin 39 dell'UART N. 2, ma lasciate il pin 39 dell'UART N. 1 allo stato logico 1. Facendo ciò l'UART N. 2 si aspetta ora di ricevere un carattere di parità dispari. L'UART N. 2, al contrario, si aspetta di ricevere un carattere di parità pari e trasmetterà un carattere asincrono di parità pari. In altre parole abbiamo chiesto deliberatamente di disaccoppiare i due UART.

Il pin 13 dovrà così posizionarsi allo stato logico 1 nel momento in cui un carattere seriale asincrono viene ricevuto dall'uno o dall'altro UART.

Passo 4

Trasmettete una qualsiasi parola di dati dall'UART N. 1 all'UART N. 2. Alla fine della trasmissione il pin 13 si posizionerà allo stato logico 1. E' vero? Rispondete si o no nello spazio seguente.

Trasmettete una qualsiasi altra parola di dati dall'UART N. 2 all'UART N. 1. Alla fine della trasmissione, il pin 13 dell'UART N. 1 dovrebbe essere allo stato logico 1. E' vero? Rispondete si o no nello spazio seguente.

Passo 5

Ora ponete a massa il pin 39 di entrambi gli UART. Trasmettete i dati dall'UART N. 1 all'UART N. 2, e dall'UART N. 2 all'UART N. 1. Il pin 13 di ciascun UART dovrebbe posizionarsi allo stato logico 0 alla fine della trasmissione di un carattere asincrono. E' vero? Rispondete si o no nello spazio seguente.

La risposta dovrebbe essere si in tutti e 3 i casi. Se avete ottenuto un tale risultato, allora avete provato con successo le funzioni del pin 13, il pin di errore di parità.

Passo 6

Come vostro test finale determinate ora la funzione del pin 15, l'ingresso di OVER-RUN. Questo ingresso va all'1 logico se il carattere inviato precedentemente non è letto, (cioé se il pin 19 non si è resettato allo stato logico 0, prima che il carattere sia trasmesso), prima che il carattere presente sia trasferito al ricevitore. In altre parole, se vi siete dimenticati di premere e rilasciare il generatore di impulsi N. 1, collegato al pin 18 del ricevitore dell'UART, prima di ricevere un nuovo carattere, allora il pin 15 si posiziona allo stato logico 1 alla fine del nuovo carattere.

Utilizzate l'UART N. 2 come trasmettitore e l'UART N. 1 come ricevitore. Premete e rilasciate il generatore di impulsi N. 1 del ricevitore. Il pin 19 del ricevitore dovrebbe posizionarsi allo 0 logico. Ora premete e rilasciate il generatore di impulsi N. 2 del trasmettitore. Un carattere seriale asincrono dovrebbe essere trasmesso dal trasmettitore al ricevitore. Alla fine della trasmissione il pin 19 del ricevitore dovrebbe posizionarsi all'1 logico e il pin 15 del ricevitore dovrebbe rimanere allo 0 logico.

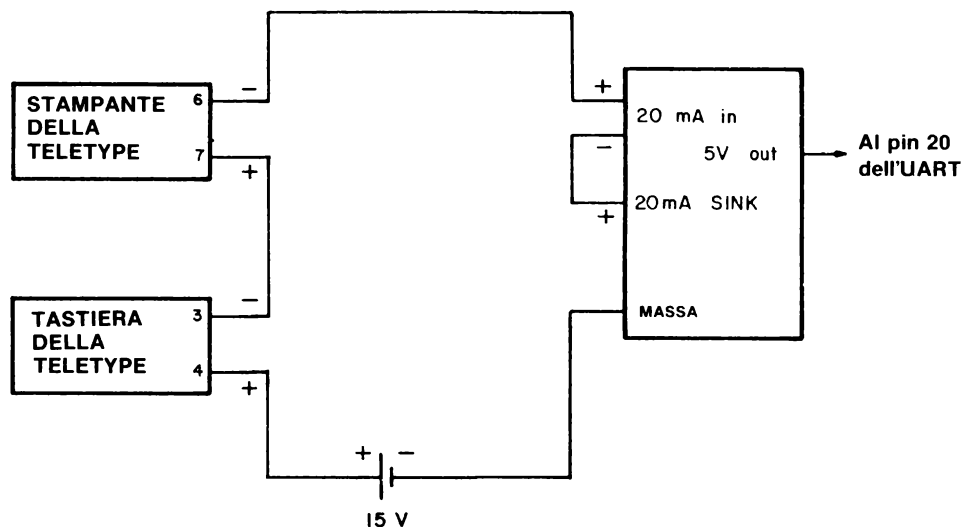
Ora premete il generatore di impulsi N. 2 del trasmettitore una seconda volta. Alla fine della trasmissione del carattere, il pin 15 dovrebbe posizionarsi allo stato logico 1 indicando che non è stata resettata l'uscita di DATA AVAILABLE al pin 19 del ricevitore prima di ricevere un nuovo carattere.

ESPERIMENTO N. 6

Scopo

Questo esperimento consente di sistemare la frequenza di clock sull'UART in modo che la frequenza di trasmissione sia 110 baud: una frequenza caratteristica per la telescrivente che si utilizza.

Schema del circuito

Parecchie cose del circuito sono le stesse di quelle mostrate nell'Esperimento N. 1. Ora anziché collegare i pin 20 e 25 insieme, farete la connessione dei pin rispettivamente ai pin "5 V OUT" e "5 V IN" di un LR-14 Digital Signal Converter Outboard. Di seguito mostriamo la rimanente parte del circuito:

Passo 1

Collegate il circuito come indicato. L'UART è collegato in modo da agire come un ricevitore di un carattere, inviato dalla telescrivente. Ponete i pin 35 e 38 di programmazione sul chip UART allo stato logico 0. In questo modo si vedranno soltanto parole di dati half-ASCII a 7 bit sul display a 7 segmenti collegato al chip UART.

Passo 2

La velocità di trasmissione dei dati di 110 baud significa che state trasmettendo 110 bit ogni secondo. Il chip UART richiede 16 impulsi di clock per trasmettere o ricevere un carattere seriale asincrono, così la velocità di 110 Baud di una telescrivente corrisponde ad una frequenza di clock sull'UART di 1760 Hz. Si deve fare in modo che la frequenza di clock sia un fattore per cento di questo valore.

Sul nostro Outboard di UART e su quello di clock, noi abbiamo osservato che una capacità connessa all'Outboard di clock ed eguale a 270 pF, produce una frequenza di clock che permetterebbe di ricevere i dati in modo corretto dall'UART. Utilizzando condensatori con valori più piccoli, varianti tra 10 pF e 200 pF, si può sistemare la frequenza a 110 baud.

Passo 3

Potete verificare se la frequenza è corretta, collegando all'intero circuito l'alimentatore, includendo il loop di corrente, e battendo i caratteri sulla tastiera della telescrivente. Se la frequenza di clock dell'UART è corretta, osserverete l'appropriato codice half ASCII (in ottale) sul display a 7 segmenti.

Alimentate l'intero sistema. Un paio di pile da +6 volt possono sostituire l'alimentatore da +15 volt. Se la telescrivente "cicaleggia", il collegamento del loop di corrente è sbagliato. Alcuni dei problemi di collegamento che abbiamo osservato con gli studenti, possono essere elencati come segue:

- Un alimentatore da +5 volt non può essere applicato all'Outboard LR-14
- Un alimentatore da +5 volt e/o − 5 volt non può essere applicato all'Outboard di UART.
- Non c'è una connessione di massa tra gli Outboard LR-14 e UART o tra l'uno e l'altro, o tra entrambi questi Outboard e il terminale negativo dell'alimentatore da + 15 volt o + 12 volt che alimenta il loop di corrente.
- I connettori della tastiera della telescrivente sono collegati in modo inverso rispetto al loop di corrente,
- I connettori della stampante della telescrivente sono collegati in modo inverso rispetto al loop di corrente,
- L'Outboard LR-14 è malfunzionante.

I problemi più comuni sono gli allacciamenti dei connettori all'alimentatore o a massa, oppure il collegamento inverso dei connettori provenienti dalla telescrivente. *Non dimenticate soprattutto di cambiare l'interruttore della telescrivente in posizione LINE piuttosto che in posizione LOCAL. In questa posizione, la telescrivente è disabilitata e non può funzionare come trasmittente e quindi nessun carattere asincrono sarà ricevuto dall'UART!*

Passo 4

Se la telescrivente non presenta problemi, premete i tasti corrispondenti al numero 1 ed alla lettera A ed osservate la visualizzazione in ottale sul display a 7 segmenti. Se appare rispettivamente 060 e 101 la frequenza di clock del chip UART varia di un valore non superiore del 5% di 110 Baud (o 1760 Hz). Se appaiono valori differenti da questi, è necessario sistemare la frequenza di clock utilizzando un ulteriore condensatore. Dopo questa verifica potete trasmettere correttamente qualsiasi lettera, numero o simbolo in codice half-ASCII. Avete così completato questo esperimento. *Conservate il vostro circuito e proseguite con il prossimo esperimento.*

ESPERIMENTO N. 7

Scopo

Questo esperimento prova le caratteristiche di un UART, che funziona come ricevitore di un carattere trasmesso da una telescrivente. Il circuito è identico a quello dato nell'esperimento precedente.

Schema del circuito

Utilizzate lo stesso circuito dell'Esperimento N. 6. I pin 25 e 38 del chip UART dovrebbero essere allo stato logico 0 per assicurarsi che soltanto un codice ottale half-ASCII apparirà sul display a 7 segmenti.

Passo 1

Premete i seguenti tasti corrispondenti a lettere, numeri, simboli, operazioni sulla telescrivente e scrivete nella colonna centrale i valori che appariranno sui display a 7 segmenti.

Lettere, numeri, simboli, operazioni	*Parola ottale a 7-bit*	*Parola ottale a 8-bit*
0	060	260
1		
4		
9		
A		
C		
G		
K		
O		
S		
U		
Z		
&		
%		
!		
.		

Passo 2

Posizionate ora i pin di programmazione 35 e 38 all'1 logico dell'Outboard UART. Ripetete il passo 1 e scrivete i valori osservati sul display, nella colonna a destra. Nello spazio seguente spiegate perché vi è una differenza tra le due colonne.

Passo 3

Con i pin 35 e 38 dell'UART ciascuno allo 0 logico, collegate il pin 13 dell'UART ad un indicatore a LED. Quando questo pin è all'1 logico ed il pin 39 è anch'esso all'1 logico, il carattere ricevuto dovrà avere parità dispari. Quando il pin 13 è allo 0 logico invece, il carattere ricevuto avrà parità pari. L'esercizio consiste nel verificare quanto detto.

Passo 4

Trasmettete le lettere e i numeri della tabella seguente, dalla telescrivente all'UART e determinate la parità di ciascun carattere trasmesso, osservando lo stato logico del pin 13 sull'UART.

Lettere o numeri	*Codice ottale ricevuto*	*Parità osservata*
0	060	
1		
2		
3		
4		
5		
6		
7		
8		
9		
A		
B		
C		
D		
E		
F		
G		

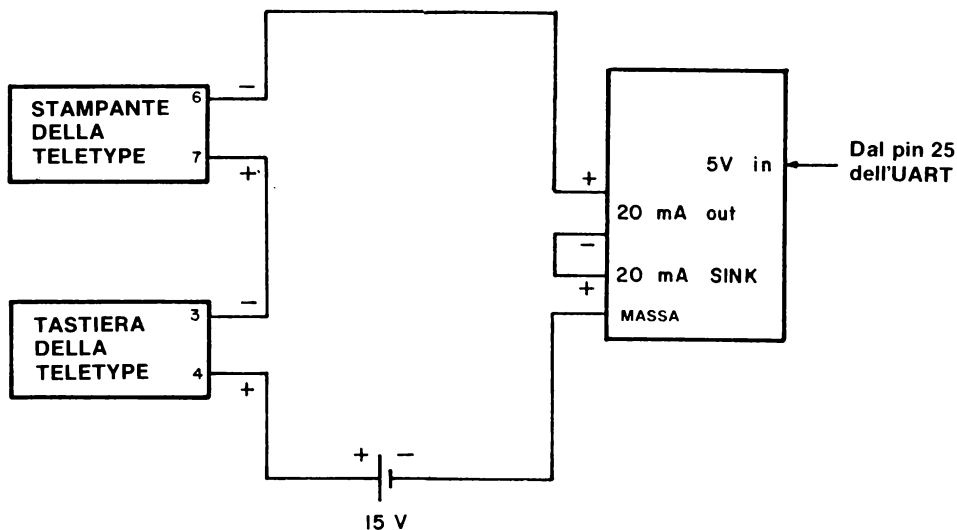
Siete soddisfatti dei risultati dell'esperimento precedente? Se no, quale è il poblema? E' possibile che l'ottavo bit dei dati che è trasmesso dalla telescrivente, piuttosto che essere il bit di parità, è sempre all'1 logico, contribuendo così al conteggio della parità? Scrivete la vostra risposta nello spazio seguente.

ESPERIMENTO N. 8

Scopo

Questo esperimento prova il funzionamento dell'UART come trasmettitore di caratteri alla telescrivente.

Schema del circuito

Il circuito è simile a quello dell'Esperimento N. 7. Effettuate i semplici cambiamenti indicati di seguito.

Passo 1

Collegate il circuito come mostrato e alimentate il sistema.

Passo 2

Premete i tasti numerici, letterali o simbolici sulla telescrivente ed osservate se appare o no il codice ottale corrispondente sul display a 7 segmenti connesso all'UART. Se le connessioni sono corrette, il display dovrebbe rimanere a 000_8. In altre parole dovreste osservare che l'UART non può funzionare come un ricevitore di un carattere trasmesso

alla telescrivente.

Passo 3

Posizionate i pin di programmazione 35 e 38 allo 0 logico. Così siete sicuri che una parola asincrona seriale a 11 bit, che l'UART trasmette alla telescrivente, conterrà un bit di start, sette bit di dati, un bit di parità e due bit di stop. Dovreste essere in grado di costruire la tabella di codici half-ASCII, come nel capitolo 5 del Bugbook I, per trasmettere lettere e numeri alla stampante della telescrivente.

Passo 4

Alimentate il sistema, cioé, entrambi il loop di corrente che l'Outboard e trasmettete i seguenti valori "settati" degli switch logici premendo e rilasciando il generatore di impulsi N. 2, come mostrato nell'Esperimento N. 1. Scrivete nello spazio seguente, nella colonna a destra, i numeri e le lettere che osservate sulla stampante.

Valori degli switch logici — *Caratteri osservati*

G	F	E	D	C	B	A	Caratteri osservati
0	1	1	0	0	0	0	
0	1	1	0	0	0	1	
0	1	1	0	1	1	1	
0	1	1	1	0	0	1	
1	0	0	0	0	0	1	
1	0	0	1	1	1	1	
1	0	1	0	0	0	0	
1	0	1	0	1	1	0	
0	1	0	0	0	1	0	
0	1	0	1	0	1	0	
0	0	0	1	0	1	0	
0	0	0	1	1	0	1	

Come fate per sapere se i precedenti risultati sono corretti? Spiegatelo nello spazio seguente.

ESPERIMENTO N. 9

Scopo

Questo esperimento prova la trasmissione in modo *half-duplex* di un carattere ASCII seriale asincrono di 11 bit tra una telescrivente e un circuito integrato UART. Nei sistemi half-duplex, un unico loop di corrente è presente, ed i caratteri possono essere trasmessi sia dalla telescrivente che dall'UART, ma non simultaneamente.

Schema del circuito

Passo 1

Eseguite i cambiamenti dei collegamenti come mostrato nel circuito sopra riportato. Come vedete vi è solo un loop di corrente.

Passo 2

Applicate tensione al breadboard ed al loop di corrente. Ricordate che una coppia di pile da +6 volt, quando collegate in serie, costituisce un buon alimentatore per il loop.

Passo 3

Questo esperimento è la combinazione degli Esperimenti N. 7 e 8. Nell'Esperimento N. 7, avete costruito un circuito per trasmettere un carattere dalla telescrivente all'UART. Nell'Esperimento N. 8, viceversa, il circuito trasmetteva un carattere dall'UART alla telescrivente. In questo esperimento trasmetterete dalla telescrivente all'UART o dall'UART alla telescrivente, cioé, potete provocare entrambe le operazioni degli Esperimenti N. 7 e 8.

Rifate l'Esperimento N. 7 e ripetete i Passi da 1 a 4. Dovreste ottenere lo stesso risultato.

Passo 4

Rifate l'Esperimento N. 8 ripetendo i Passi 3 e 4. Dovreste ottenere lo stesso risultato di quell'Esperimento.

Passo 5

Rimane per ultimo il caso in cui si voglia trasmettere simultaneamente dall'UART e dalla telescrivente. Questo presenta alcune difficoltà.

Nel nostro caso abbiamo tentato di trasmettere 012_8 dall'UART e 060_8 dalla telescrivente, premendo il tasto corrispondente al numero decimale 0 e, nello stesso tempo, rilasciando in generatore di impulsi di DATA STROBE N. 2. I caratteri stampati sulla telescrivente e osservati sul display a 7 segmenti sono i seguenti:

Caratteri stampati	*Caratteri visualizzati*
X	170
0	141
0A	141
blank	070
0A	141
blank	040

E' difficile trasmettere simultaneamente dall'UART e dalla telescrivente. Il risultato di questa prova è inutile.

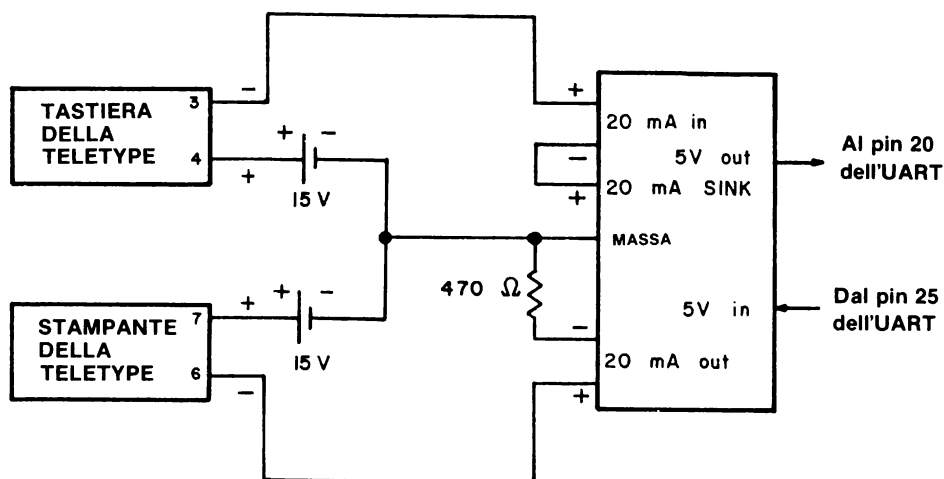
Convincetevi comunque, che non si può trasmettere da due trasmettitori contemporaneamente quando c'è un unico loop di corrente.

ESPERIMENTO N. 10

Scopo

Questo esperimento prova la trasmissione in modo full-duplex, di un carattere ASCII seriale asincrono di 11 bit, tra una telescrivente e un circuito integrato UART. I sistemi full duplex, sono costituiti da due loop di corrente e il carattere può essere simultaneamente trasmesso o ricevuto sia dalla telescrivente che dall'UART.

Schema del circuito

Il circuito è simile ai circuiti degli Esperimenti N. 6, 7, 8 e 9.

Passo 1

Eseguite i cambiamenti nel circuito seguendo lo schema della figura precedente. Notate che esistono due loop di corrente, e che è stato inserito un resistore di 470 Ω per sistemare il livello della corrente di uno dei due loop. Alimentate ciascun loop in modo indipendente.

Passo 2

Applicate tensione al breadboard e ai due loop di corrente. Una coppia di pile sostituiscono l'alimentatore a +15 volt.

Passo 3

Posizionatè l'interruttore della telescrivente su LINE. Se la telescrivente "cicaleggia", non fluisce corrente nel loop di corrente della tastiera. Dovrete allora fare le opportune modifiche nel collegamento.

Battete il tasto corrispondente al numero 0 sulla telescrivente. Ora, per la prima volta, la stampante non stamperà il carattere perché non c'è corrente nel loop di corrente della tastiera. Osserverete comunque, il numero ottale 060_8 sui 3 display a LED a 7 segmenti connessi all'UART.

Passo 4

Posizionate gli switch connessi all'UART a GFEDCBA = 0110000. Premete e rilasciate il generatore di impulsi N. 2 di DATA STROBE, e spiegate ciò che accade nello spazio seguente:

Nel nostro caso, abbiamo osservato la stampa sulla telescrivente del numero 0.

Se posizionate invece gli switch a 0001010 = GFEDCBA e premete e rilasciate il generatore di impulsi di DATA STROBE, che cosa accade? Spiegatelo di seguito.

Nel nostro caso, è stata attuata un'operazione di LINE FEED. Il codice ASCII a 7 bit, corrisponde ad un LINE FEED, 012, non a apparirà sui tre display a 7 segmenti.

Passo 5

Ora potete capire che cosa accade in questo circuito:

- L'UART agisce come un trasmettitore alla stampante della telescrivente che riceve i caratteri trasmessi e li stampa sulla carta.
- La telescrivente agisce come un trasmettitore al display sul chip UART.

Quello che è particolarmente interessante a proposito del circuito, è che sia la telescrivente, che il chip UART possono trasmettere simultaneamente informazioni. Potete rilevarlo rilasciando il generatore di impulsi del DATA STROBE, e contemporaneamente premere un carattere sulla telescrivente. Le due azioni non interferiscono tra loro. Se il chip UART è stato connesso a strumenti di laboratorio, potete mandare informazioni allo strumento, nello stesso momento in cui i dati dallo strumento arrivano sulla telescrivente!

Mr. David G. Larsen è un istruttore del Dipartimento di Ingegneria Chimica del Virginia Polytechnic Institute & State University dove svolge attività didattica a vari livelli nel campo dell'elettronica analogica e digitale. È coautore di altri Bugbook e di una pubblicazione mensile, denominata Columns, riguardante l'interfacciamento dei microcomputer. Con il dr. Rony cura una serie di corsi sui microcomputer, sotto gli auspici della Extension Division della suddetta Università. Questi corsi sono particolarmente apprezzati e seguiti da professionisti di ogni parte del mondo.

Il **dr. Peter R. Rony** è professore presso il Dipartimento di Ingegneria Chimica del Virginia Polytechnic Institute & State University (USA). L'elettronica digitale ed i microcomputer giocheranno un ruolo molto importante nei controlli di processo, soggetto questo di considerevole interesse per l'ingegneria chimica. Il dr. Rony è coautore di molti altri Bugbooks e di una pubblicazione mensile, denominata Columns, riguardante l'interfacciamento dei microcomputer, che appare su *American Laboratory, Computer Design, Ham Radio Magazine,* in Usa, *Electroniker* in Germania e *Elettronica Oggi* in Italia.

L. 4.500
(4.245)

Edizione Italiana del "the BUGBOOK IIA"